Die nachstehende Serie beschreibt ein Mikrocomputersystem, das modular aufgebaut ist. Grundsätzlich ist dieses System nicht an bestimmte Mikroprozessortypen gebunden; als erstes Beispiel wird ein System mit der 8080-Familie beschrieben, weil dies heute der am häufigsten verwendete Typ ist. Der modulare Aufbau gewährleistet eine nahezu beliebige Erweiterbarkeit. Es werden für den Praktiker zur Einarbeitung zunächst einfache Anwendungen mit elektronischen Unterhaltungsspielen beschrieben. Das Ganze wird praxisnah – in der Art eines Kochbuches – aufgebaut.
Beim Vergleich dieser Anwendungen mit bisherigen Digitalschaltungen wird automatisch das Verständnis für die Materie der Mikrocomputer und Mikroprozessoren geweckt. Man erkennt die Vorteile und Grenzen dieser neuen Technik.

Horst Pelka

Praxis mit Mikroprozessoren

1. Teil

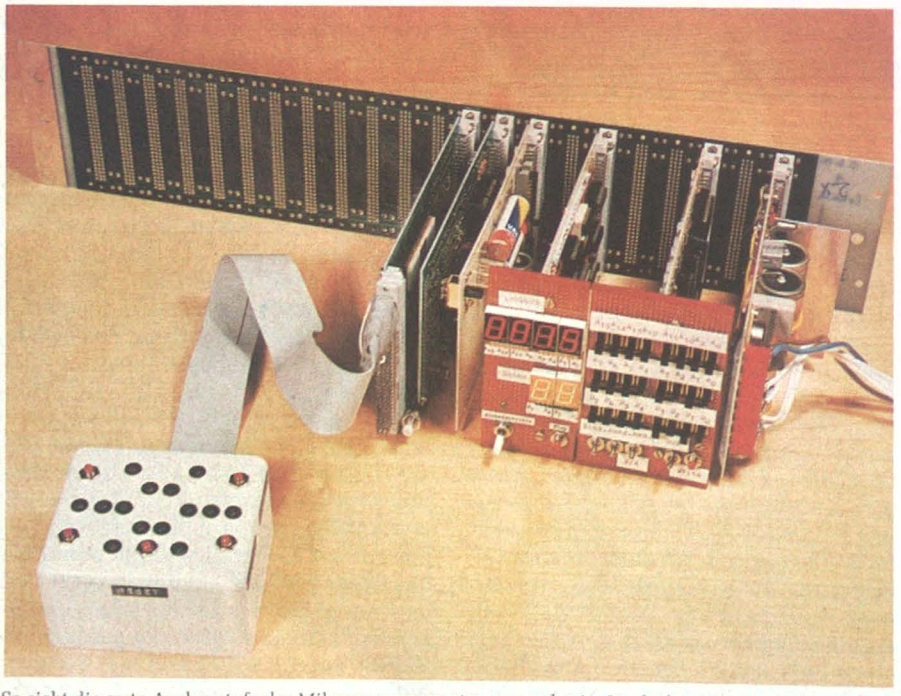
In jüngster Zeit eroberte sich eine spezielle Gruppe integrierter Bausteine einen festen Platz in der Elektronik: die Mikroprozessoren. Es handelt sich um die konsequente Weiterentwicklung hochintegrierter Digitalbausteine, die logische Operationen nach einem festzulegenden Ablauf ausführen. Unter den Operationen sind logische Verknüpfungen, Vergleiche, Additionen und Subtraktionen zu verstehen. Betrachten wir eine herkömmliche Digitalschaltung (*Bild 1*), so läßt sich diese in drei Funktionsblöcke zerlegen.

Der Kern dieser Digitalschaltung ist eine festverdrahtete Logik, die eine einmal festgelegte Funktion ausführt. An diese Logik sind ein Bedienteil sowie Geber und Sensoren angeschlossen. Je nach Anwendungsfall ist der Aufwand dieser beiden peripheren Blöcke kleiner oder größer. In speziellen Fällen, so z. B. bei Unterhaltungsspielen, ist der Funktionsteil *Geber und Sensoren* im eigentlichen Bedienungsteil eingeschlossen. Es sind Tasten und Anzeigeelemente vorhanden, mit denen der Mensch mit der Logik kommunizieren kann.

Nehmen wir als Beispiel das in der ELO [1] beschriebene elektronische Spiel „Seeschlacht", so enthält der Bedienteil fünf Tasten und einen Einschalter sowie zwölf Anzeige-LED's. Die festverdrahtete Logik besteht dann aus zehn Gatterbausteinen, einem Dezimalzähler und einem BCD-Dezimaldecoder sowie Transistoren, Widerständen und Kondensatoren.

Die gleiche Aufgabe läßt sich mit einem Mikrocomputersystem (*Bild 2*) lösen. Statt der festverdrahteten Logik ist hier der Mikrocomputer eingesetzt; an ihn ist ein Bedienteil mit Tasten und Anzeigelampen angeschlossen und gegebenenfalls ein Funktionsblock *Geber und Sensoren*. Oberhalb des Blockes Mikrocomputer finden wir noch die Funktionsgruppe *Anzeigemodul* und *Modul für direkten Speicherzugriff*. Diese beiden Moduln sind für den eigentlichen Funktionsablauf nicht notwendig, gestatten aber einen Einblick in die inneren Vorgänge des Mikrocomputers sowie eine Programmierung des RAM-Speicherbereiches.

Im Gegensatz zur vorher genannten festverdrahteten Logik arbeitet der Mikrocomputer mit einer programmierbaren Logik. Wir alle kennen das altbekannte Programmschaltwerk in einer vollautomatischen Waschmaschine. Es handelt sich um eine programmgesteuerte Logik, deren Programm festgelegt ist. Wenn man nun noch die Möglichkeit einschließt, dieses Programm ändern zu können, kommen wir zur programmierbaren Logik. Im Gegensatz zur festverdrahteten Logik, bei der viele Abläufe parallel geschehen können, arbeitet eine programmgesteuerte und programmierbare Logik seriell, d. h. jeder Schritt erfolgt einzeln nach dem anderen. Die Art der logischen Schritte, in welcher Reihenfolge sie erfolgen und was dabei passieren wird, wird durch einen sogenannten Programmspeicher bestimmt.

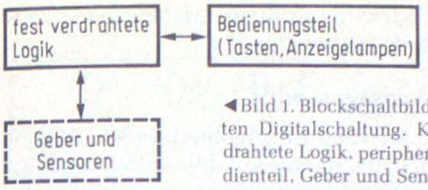


So sieht die erste Ausbaustufe des Mikroprozessorsystems aus, das im Laufe dieses Lehrgangs aufgebaut werden kann

Begriffsbestimmung

Ein Mikrocomputer besteht aus einem Mikroprozessor und mehreren zusätzlichen Bausteinen, die seinen Programmablauf speichern, den Kontakt zur „Außenwelt" herstellen. Sie sind in Bild 2 als Speichermodul und Ein-Ausgabemodul gekennzeichnet. Die Verbindung zwischen den einzelnen Bausteinen innerhalb des Mikrocomputers geschieht über sogenannte Bus-Systeme. Darunter versteht man Sammelschienen mit bis zu 16 Leitungen, an denen sämtliche Bausteine parallel angeschlossen sind. Im wesentlichen unterscheidet man drei Arten von Sammelschienen (Busse):

- Adreßbus,
- Datenbus,
- Steuerbus.



◀ Bild 1. Blockschaltbild einer festverdrahteten Digitalschaltung. Kern ist die festverdrahtete Logik, periphere Elemente sind Bedienteil, Geber und Sensoren

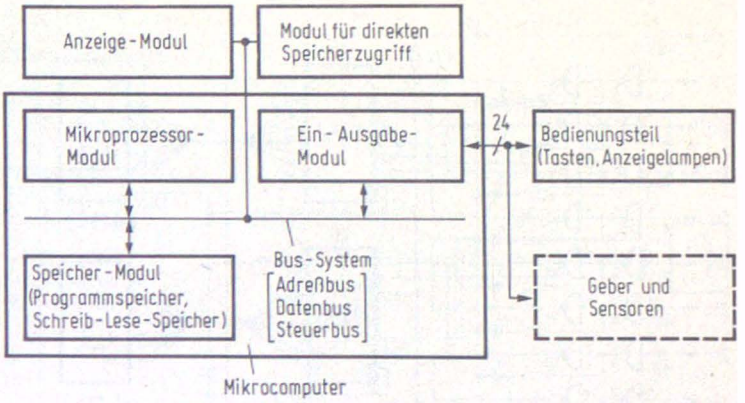


Bild 2. ▶
Aufbau eines Mikrocomputersystems. Peripherie: Bedienteil, Geber und Sensoren. Hilfsmodule: Anzeige und direkter Speicherzugriff

Über den *Adreßbus* werden die Adressen der einzelnen Speicherelemente und Ein- und Ausgabe-Kanäle übertragen. Adreßgeber ist in jedem Fall der Mikroprozessor, welcher die Adressen des Programmspeichers der Reihe nach ausgibt. Werden im Programmablauf andere Speicherstellen, z. B. die des Schreib/Lesespeichers (RAM) oder Ein- und Ausgabeadressen, angesprochen, so übernimmt der Mikroprozessor die Vermittlung der entsprechenden Adressen. Sie stehen ursprünglich im eigentlichen Anwenderprogramm, welches im Programmspeicher aufbewahrt ist.

Über den *Datenbus* werden die einzelnen Daten übertragen, je nach Wortbreite des Systems kann der Datenbus vier bis sechzehn Bit umfassen. Über den Datenbus werden die Daten vom Programmspeicher zum Mikroprozessor, die Daten von den Ein- und Ausgängen zum Mikroprozessor und zurück sowie die Daten vom Schreib/Lesespeicher zum Mikroprozessor und zurück übertragen.

Über den *Steuerbus* werden die einzelnen Signale übertragen: Ob z. B. ein Speicher gelesen wird oder etwas in den Speicher eingeschrieben wird, ob der Ein/Ausgabebaustein Daten ein- oder ausgibt. Einige Leitungen des Steuerbusses sind nach außen geführt, so z. B. die Rücksetzleitung sowie Leitungen, mit denen der Ablauf des Mikroprozessors angehalten oder ganz gestoppt werden kann. Diese Leitungen braucht man, wenn man z. B. von außen direkt Speicherstellen lesen will oder Informationen in den Speicher direkt eingeben möchte.

System-Aufteilung

Das hier beschriebene System ist, wie schon gesagt, modular aufgebaut. Jede Funktion ist auf einer eigenen Steckkarte untergebracht.

Der Mikroprozessor übernimmt die Koordination aller Funktionsgruppen und steuert sämtliche internen Abläufe. Das in ihm enthaltene Rechenwerk ALU führt entsprechend den Programmanweisungen die logischen Operationen mit dem Operanden durch. Operanden sind beispielsweise die über den Ein/Ausgabemodul eingehenden Daten oder im Programm selbst enthaltene Konstanten. In den Speichermodul können sowohl ROM-Speicher (Read only memory), PROM-Speicher (programmable read only memory) als auch RAM-Speicher (Schreib/Lesespeicher) eingesetzt werden. Wie wir noch später bei der Beschreibung des eigentlichen Speichermoduls sehen werden, geschieht die Bestückung je nach den Möglichkeiten, die dem Leser für die Programmierung von Speichern zur Verfügung stehen.

Über separate Ein/Ausgabebausteine erfolgt die Anpassung der Informationen des Mikrocomputers an den peripheren Aufbau. Im Ein/Ausgabemodul können die Ausgangsdaten zwischengespeichert werden, so daß der Bedienteil den eigentlichen seriellen Ablauf des Mikrocomputers nicht merkt und gar keinen Unterschied zu einer festverdrahteten Logik findet.

Erwähnt soll noch werden, daß es heute bereits Mikrocomputer gibt, die auf einem einzigen Halbleiterchip untergebracht sind.

Modular-System

Bei den hier beschriebenen Mikrocomputersystemen handelt es sich um ein Modularsystem, das durch die Art seiner Struktur beliebig erweiterbar ist. Als Mikroprozessor wird der weitverbreitete Typ 8080 angewendet; später wird auch eine Platine mit dem Nachfolgertyp beschrieben werden. Die Steckerbelegung *(Tabelle)* ist identisch mit dem SMP 80-System von der Firma Siemens, welches im Europakartenformat aufgebaut ist. Damit ist auch dem Leserkreis, der sich nicht alle Platinen selbst erstellen möchte, die Möglichkeit geboten, einige Platinen fertigbeschaltet und geprüft kaufen zu können. Die in der FUNKSCHAU beschriebenen Platinen enthalten nur den für die ersten Anwendungsbeispiele unbedingt notwendigen Bausteineaufwand.

Das ganze System besteht aus

1. Anzeigemodul
2. Modul für direkten Speicherzugriff
3. Speichermodul
4. Mikroprozessormodul
5. Ein/Ausgabemodul
6. Busverdrahtungsplatine
7. Netzteil
8. Bedienteil.

Der Anzeigemodul und der Modul für direkten Speicherzugriff werden für

Tabelle der Steckerbelegung der Rückwandverdrahtung aller Moduln

	a	c
1	–15 V	–12 V
2	– 5 V	0 V
3	Φ1TTL	OSC
4	Φ2TTL	$\overline{\text{MMI/0}}$
5	STSTB	A12
6	$\overline{\text{RESET}}$	A0
7	SYNC	A13
8	$\overline{\text{MEMR}}$	A1
9	$\overline{\text{RESIN}}$	A14
10	$\overline{\text{MEMW}}$	A2
11	WAIT	A15
12	RDYIN	A3
13	BUSEN	ZF0
14	DBO	A4
15	HLDA	ZF1
16	DB1	A5
17	$\overline{\text{HOLD}}$	ZF2
18	DB2	A6
19	$\overline{\text{INT}}$	ZF3
20	DB3	A7
21	$\overline{\text{INTE}}$	ZF4
22	DB4	A8
23	$\overline{\text{INTA}}$	ZF5
24	DB5	A9
25	$\overline{\text{CASO}}$	ZF6
26	DB6	A10
27	$\overline{\text{CAS1}}$	ZF7
28	DB7	A11
29	$\overline{\text{CAS2}}$	ZF8
30	$\overline{\text{IOW}}$	$\overline{\text{IOR}}$
31	+15 V	0 V
32	+5 V	+12 V
	64polig	

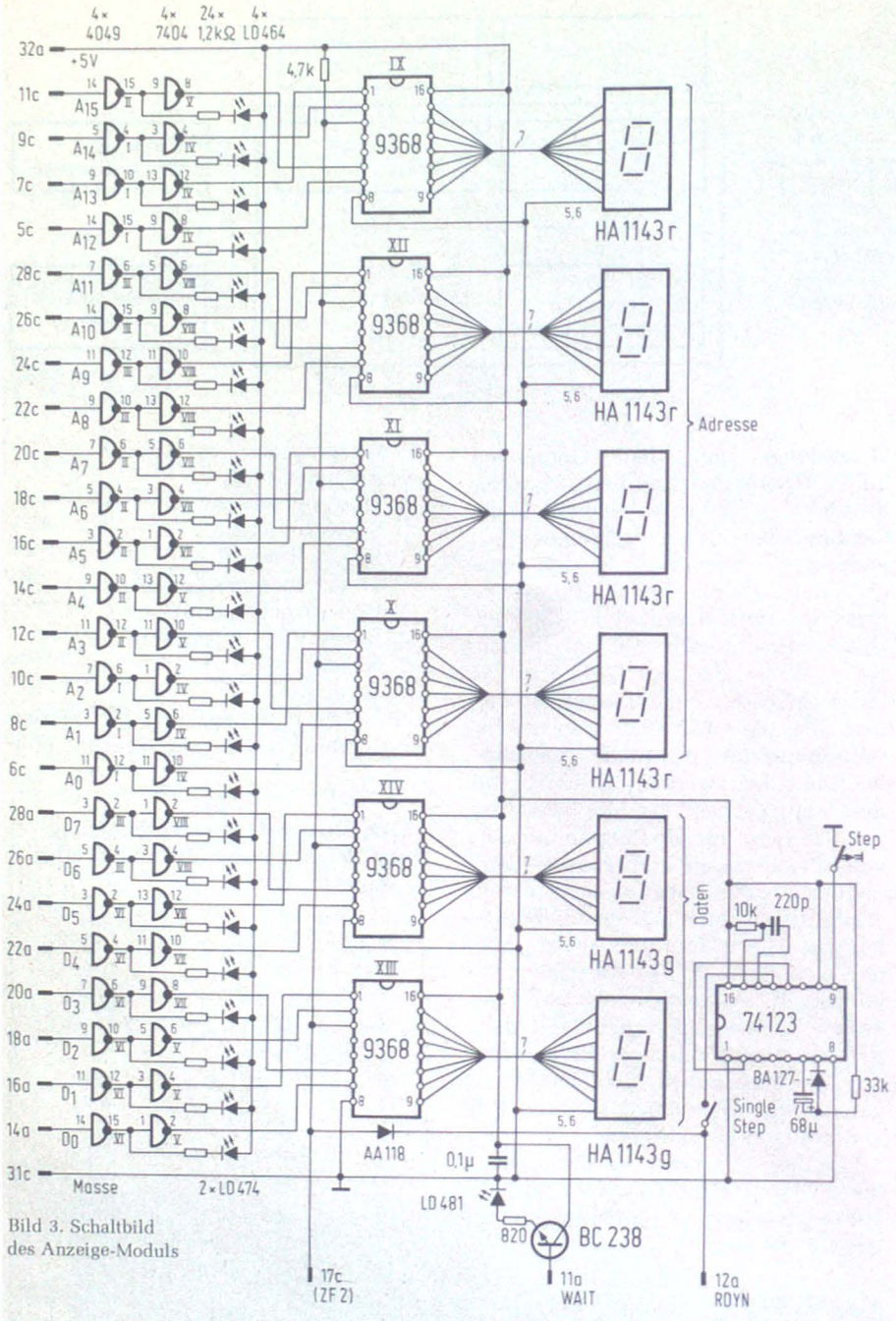


Bild 3. Schaltbild des Anzeige-Moduls

den eigentlichen Betrieb nicht benötigt. Der Verfasser geht aber von der Voraussetzung aus, daß dem Leser keinerlei Hilfsmittel zur Programmierung zur Verfügung stehen und er das Programm mit eigenen Mitteln in den Speichermodul einbringen muß. In diesem Falle benötigt er jedoch die beiden Moduln.

Für diejenigen Leser, die schon einige Mikroprozessorkenntnisse besitzen und das Programm an einem Entwicklungsplatz assemblieren und anschließend über ein PROM-Programmiergerät in ein PROM einbringen können sowie das System über einen Echtzeit-Adapter an einen Entwicklungsplatz anschließen können, können sich den Bau der beiden Moduln *Anzeigemodul* und *Modul für direkten Speicherzugriff* ersparen.

Aus didaktischen Gründen werden zuerst der Anzeigemodul und der Modul für direkten Speicherzugriff beschrieben. Dabei kann sich der Leser in die Codierung in Binär- und Hexadezimalcode einarbeiten [3].

Anzeigemodul

Mit dem Anzeigemodul (*Bild* 3) können die Signale auf dem Adreß- und Datenbus angezeigt werden. Gleichzeitig ist auf dieser Platine eine Hilfsschaltung untergebracht, mit der man jeden einzelnen Schritt des Mikroprozessors verfolgen kann.

Die Darstellung des Pegels jeder einzelnen Leitung erfolgt mit Leuchtdioden (in Bündeln zu je vier Leitungen). Jedes Viererbündel wird zu einer Sedezimalzahl dekodiert (Hex code) und mit einer LED-Siebensegment-Anzeige angezeigt. *Bild* 4 zeigt den Zusammenhang zwischen dem Dualcode und dem Hexcode.

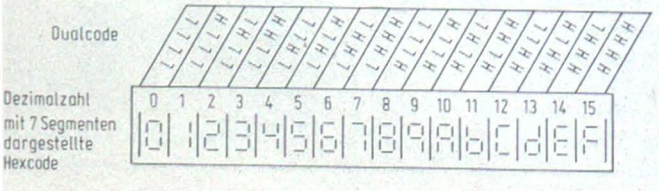
Der Adreßbus hat eine Wortbreite von 16 Bit. Zu seiner Anzeige werden also vier LED-Anzeigen bzw. vier Zeilen zu je einer LED benötigt. Der Verfasser hat im Mustergerät jeweils eine Anzeige und eine Viererzeile übereinander angeordnet, so daß man den Zusammenhang zwischen dem Dual- und Hexcode sofort erkennen kann.

Der Datenbus hat eine Breite von 8 Bit = 1 Byte. Dies entspricht zwei LED-Anzeigen und zwei Zeilen mit je vier LED-Punkten.

Der Anzeigemodul läßt sich für jede Art von Mikroprozessorsystemen verwenden, die mit TTL-Pegel arbeiten. Um die Adreß- und Datenbusse nicht zu belasten, sind am Eingang des Anzeigemoduls invertierende Hexpuffer in CMOS-Technik vorgesehen. Die Ausgänge der Hexpuffer führen dann über Widerstände von 820 Ω auf die Leuchtdioden der LED-Zeilen. Die zweiten Anschlüsse der Leuchtdioden sind mit +5 V verbunden. Bild 5 zeigt die Anschlußfolge der Hexpuffer-Bausteine.

Wegen der größeren Ausgangsbelastbarkeit des Ausgangs der Hexpuffer im L-Zustand wurden invertierende Puffer gewählt. Wer Geld sparen möchte, kann in einer ersten Ausbaustufe mit diesen Hexpuffern und nachgeschalteten LED-Zeilen zur Anzeige der Bus-Pegel auskommen. Er muß die Umwandlung in Hexcode dann selbst gedanklich vornehmen.

Da der Binär-Hex-Dekoderbaustein 9368 an den Eingängen H-Pegel für eine



◀ Bild 4. Zusammenhang zwischen Dualcode und Hexcode

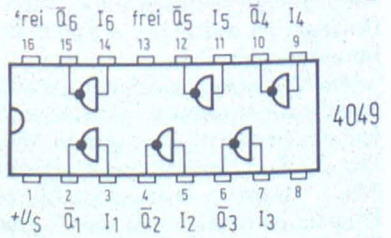


Bild 5. ▶ Anschlußanordnung Hex-Inverter-Treiberstufen HEF 4049 (Ansicht von oben)

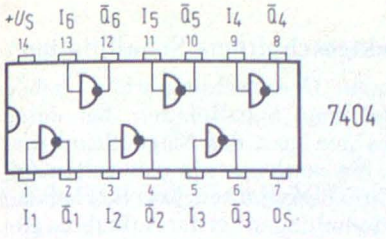


Bild 6. Anschlußanordnung der sechsfachen Inverter 7404 (FLH 2117) (Ansicht von oben)

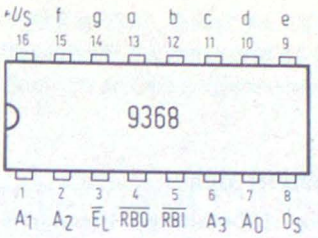
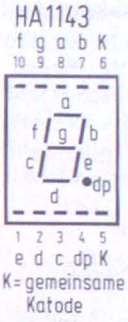


Bild 7. Anschlußanordnung der Hexadezimal-7-Segment-Dekoder-Treiber 9368 (Ansicht von oben)



◀ Bild 8. Anschlußanordnung der LED-7-Segment-Anzeigen HA 1143 (Draufsicht – Displayseite)

„1" benötigt, sind zwischen den Ausgängen der CMOS-Hexpuffer und den Eingängen der 7-Segment-Decoderbausteine noch einmal TTL-Hexinverter 7404 geschaltet. *Bild 6* zeigt die Anschlußfolge der TTL-Hexinverter. Es können aber ebensogut Bausteine in TTL-Lowpower-Schottky-Technik oder auch pinkompatible CMOS-Bausteine eingesetzt werden. Bei nicht pinkompatiblen CMOS-Bausteinen muß die Anschlußfolge entsprechend berücksichtigt werden.

Es folgen die Hexadezimal-7-Segment-Dekoder an deren Ausgänge die LED-Anzeigen HA 1143 angeschlossen sind. *Bild 7* zeigt die Anschlußfolge des Hexadezimal-7-Segment-Dekoders, *Bild 8* die der LED-Anzeigen.

Die Segmente der LED-Anzeigen sind direkt mit dem Decoderbaustein verbunden. Der Decoderbaustein hat eine eingebaute Strom-Begrenzung an den Ausgängen, so daß man dort auf Strombegrenzungswiderstände verzichten kann.

Die Anschlüsse 3 der Dekoderbausteine sind alle miteinander verbunden und führen auf Anschluß 17 c des Platinensteckers. Zur Freigabe der Anzeige muß diese Leitung auf L-Pegel liegen. Dies geschieht bei Abfrage mit Einzelschritten über die Diode AA 118. Bei anderen Anwendungen, z. B. wenn einzelne Adressen aufgetastet werden sollen, wird über den Anschluß 17 c ein L-Impuls zugeführt. Der Widerstand 4,7 kΩ nach +5 V sorgt für einwandfreien H-Pegel, wenn die Anzeige gesperrt sein soll.

Bild 9 zeigt noch einmal die Möglichkeiten der Anschaltung von LED-Zeilen und Hex-7-Segment-Dekodern an die Busleitungen. Ganz oben ist die in Bild 3 gezeigte Anschaltung, darunter eine Schaltung unter Verwendung des CMOS-Bausteines 70 C 95, welcher bei H-Ausgangspegel stärker belastbar ist. Die darunter liegende Schaltung arbeitet mit dem nicht invertierenden Pufferbaustein 4050, dafür muß den LED-Zeilen ein Treibertransistor BC 238 vorgeschaltet werden. Bei der vorletzten Schaltung wird ein Trenntransistor BC 238 C mit hoher Stromverstärkung als Puffer benutzt. Hierbei werden allerdings die Busleitungen bei H-Pegel mit Strom belastet.

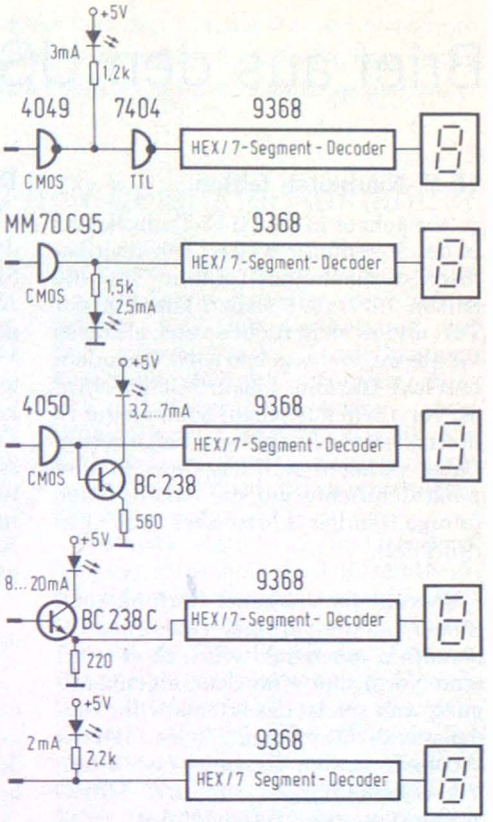


Bild 9. ▶ Verschiedene Möglichkeiten der Anschaltung von LED-Zeilen und Hexadezimal-7-Segment-Dekodern an den Busleitungen

Die Schaltung ist also nur anwendbar, wenn die Busse im Mikroprozessorsystem bereits durch Pufferstufen verstärkt wurden. Die untere Schaltung belastet die Busse bei L-Pegel mit etwa 2 mA für die LED-Zeilen. Hinzukommen noch einmal bis zu 1,6 mA für den 7-Segment-Dekoder. Letztere Schaltung ist also nur brauchbar, wenn die Busse stark belastbar ausgeführt sind.

Auf der Platine des Anzeigemoduls befindet sich noch eine Hilfsschaltung, mit der man den Ablauf des Mikroprozessors mit Einzelschritten steuern kann. Zur Steuerung des Mikroprozessors wird die Steuerbusleitung RDYN herangezogen. Diese Leitung wird normalerweise in Mikroprozessorsystemen bei Anschaltung von langsamen Speichern gebraucht. Solange die Leitung RDYN L-Pegel führt, hält der Mikroprozessor an und wartet, bis der Speicher durch H-Pegel auf dieser Leitung ihm anzeigt, daß er fertig ist. Diese Funktion kann man natürlich auch für eine manuelle Steuerung des Mikroprozessors ausnutzen. Mit dem Schalter „Single-step" wird die Leitung RDYN mit dem Ausgang des Anschluß 13 des Doppel-Mono-Flops 74123 verbunden (*Bild 10*). Dieser Anschluß führt im Ruhezustand L-Pegel. Beim Drücken der Taste „Step" entsteht durch das RC-Glied 33 kΩ/68 µF ein Impuls von etwa 1 s Länge. Die Rückflanke dieses Impulses triggert den zweiten Mono-Flop, welcher daraufhin einen Impuls von etwa 600 ns abgibt. Dieser kurze Impuls über die Leitung RDYN genügt beim Mikroprozessorsystem 8080 für einen Anlauf und einen Zyklus bis zur nächsten Adresse.

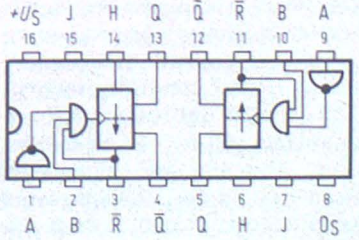


Bild 10. Anschlußanordnung der nachtriggerbaren Kippstufe 74 123 (FLK 121) (Ansicht von oben)

Die Rückmeldung vom Mikroprozessor erfolgt über die Leitung „WAIT". Sie führt H-Pegel, wenn der Mikroprozessor sich im Wartezustand befindet. Dieser H-Pegel läßt über den Puffertransistor BC 238 die LED LD 481 aufleuchten. *(Fortsetzung folgt)*

Literatur

[1] Ruff: „Seeschlacht", ein elektronisches Unterhaltungsspiel für zwei Personen ELO 1 (1975) 3, Seite 16–18

[2] Pelka: Was ist ein Mikroprozessor? RPB 82 Franzis-Verlag, München

Horst Pelka

Im zweiten Teil dieser Reihe beschreibt der Autor den Aufbau der Anzeige mit Hilfe der Fädeltechnik. Anschließend geht er auf die Funktion des Moduls für den direkten Speicherzugriff ein.

Praxis mit Mikroprozessoren

2. Teil

Aufbau

Der Aufbau der Platine in Europaformat 160 mm x 100 mm² erfolgte auf einer Grundplatte mit Lochraster 2,5 mm. Die Verdrahtung wurde in Fädeltechnik ausgeführt.

Die Fädeltechnik dient zur Realisierung von Schaltungen mit integrierten und auch diskreten Bauelementen. Der Vorteil der Fädeltechnik besteht in der schnellen Verwirklichung von Schaltungsaufbauten ohne Verwendung geätzter Leiterbahnen. Die Fädelgrundplatte trägt ein vorgebohrtes Lochraster. Der Lochdurchmesser beträgt 1,3 mm bei einem Rasterabstand von 2,5 mm. Auf die Verdrahtungsseite der Fädelplatte werden sogenannte Fädelkämme gesteckt, die zur Führung der Drähte dienen.

Die Drähte verbinden Anschlußleiste und Bauteile miteinander. Die Speisespannungen können über Stromschienen auf der Bauelementeseite zugeführt werden.

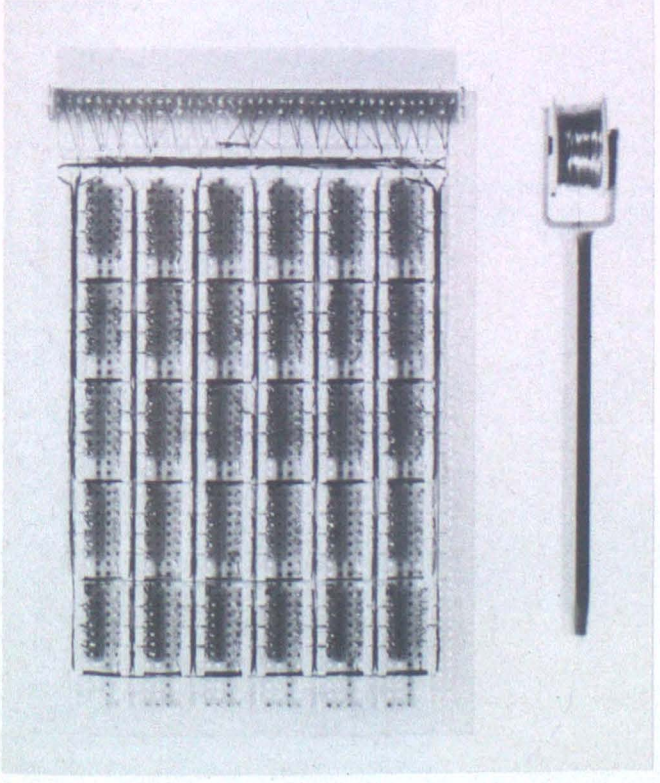
Der Fädelstift trägt eine Rolle mit lackisoliertem, lötbarem Cu-Draht. Die Isolierschicht schmilzt beim Lötvorgang. Von der Rolle läuft der Draht durch den Fädelstift, tritt an der Stiftspitze aus und wird an einen gewünschten Anschluß geführt.

Dies kann z. B. ein Stift der Anschlußleiste sein, an dem der Draht zweckmäßigerweise angelötet wird. Von hier aus kann zu beliebigen Anschlußpunkten hin gefädelt werden, wobei der Draht lediglich mit ein paar Windungen um den erreichten Stift des jeweiligen Bauteils gewickelt, anschließend weitergeführt bzw. abgeschnitten wird.

Bild 11 zeigt eine in Fädeltechnik beschaltete Platine, Bild 12 den fertigen Anzeigemodul. Die Frontplatte wurde

Einige technische Daten der Fädeltechnik:

Drahtart: doppelt isolierter lötfähiger Lackdraht	Cu-LL rt V1 Typ 28 F
Drahtdurchmesser	0,2 mm
Dicke der Isolationsschicht	13 µ
Spannungsfestigkeit	650 V (50 V/µIsolationsdicke)
Drahtwiderstand	1 Ω/1,8 m
Drahtrolle, bewickelt	50 m
Rastermaß der Fädelgrundplatte	2,5 mm
Durchmesser der Rasterbohrungen	1,3 mm
Thermische Dauerbelastung	5 A/mm², Gleichstrom
Störsicherheit	0,4 V/8 cm Drahtparallel-Führung, max. zul. Frequenz 3,2 MHz
Grundplattenmaterial	Hartpapier, TFL 1,5 DIN 40605-HP 2063 W-GE Epoxid, TFL 1,5 DIN 40606-HGW 2372



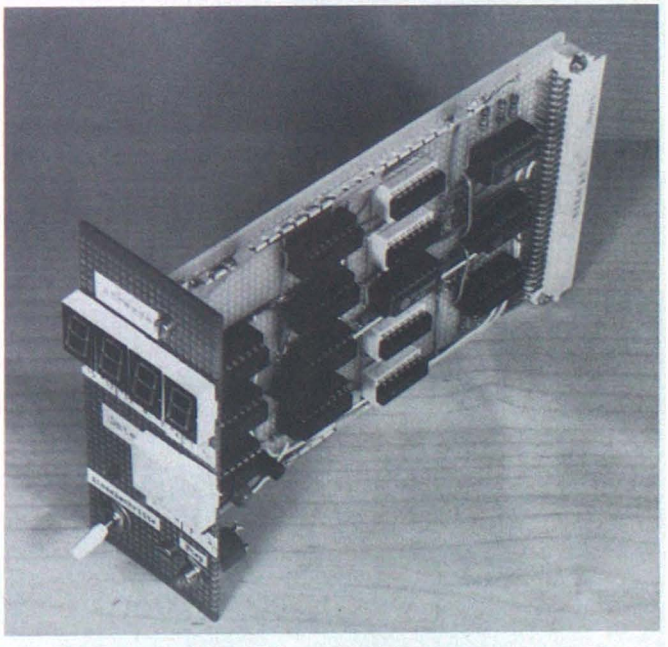


Bild 12. Foto des fertigen Anzeige-Moduls

◀ Bild 11. Beschaltungsseite einer mit integrierten Schaltungen bestückten Fädelplatte. Die Drähte werden in Fädelkämmen geführt. Rechts daneben der Fädelstift

Produktbezeichnung	Abmessungen Stück je Packung	Bestell-Bezeichnung	Lager-Nr.
Fädelgrundplatte Hartpapier-Ausführung	100 x 160 mm 1 Stück	C26117-A268-C1	5908
Fädelgrundplatte Epoxyd-Ausführung*)		DKBE26601-68	5909
Fädelgrundplatte Hartpapier-Ausführung	110 x 160 mm 1 Stück	C26117-A270-C1	5910
Fädelgrundplatte* Epoxyd-Ausführung		DKBE26601-72	5911
Stromschiene	30 Stück	C26117-A268-C2	5912
Fädelkamm (Kerbe rund)	Kammlänge 120 mm 30 Stück	C26117-A268-C4	5913
Fädelkamm (Kerbe schwalbenschwanzförmig)	Kammlänge 98 mm 30 Stück	C26117-A270-C4	5914
Fädelstift	1 Stück	C26407-A36-A1	5915
Drahtrolle für Fädelstift (bewickelt) rot		C26407-A36-C2 rot	5916
grün	Drahtlänge 50 m	C26407-A36-C2 grün	5917
blau	Rolle	C26407-A36-C2 blau braun	5918

*) für höhere mechanische Beanspruchung

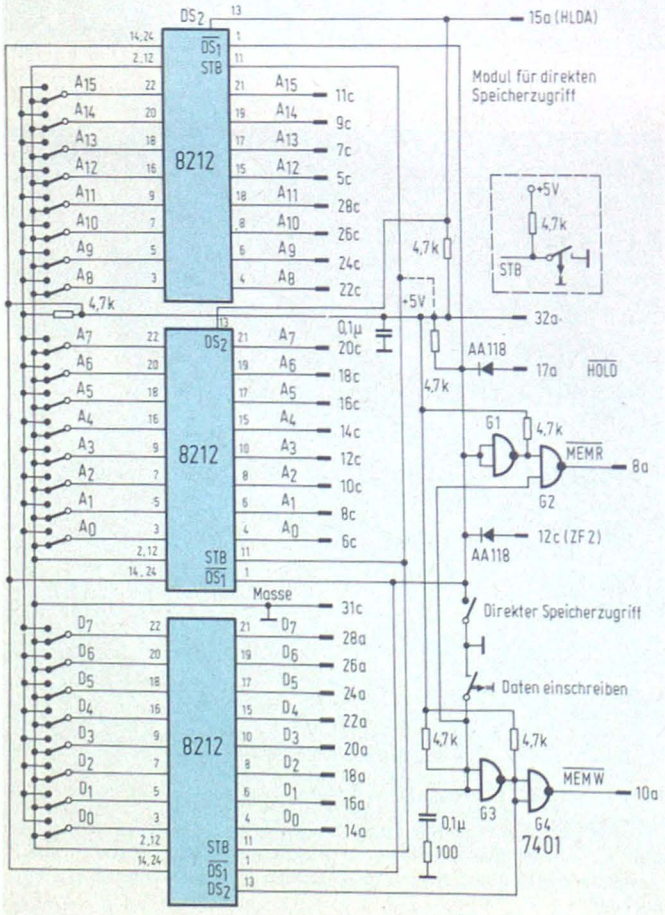


Bild 13. Schaltbild des Moduls für direkten Speicherzugriff

von einer 110 m breiten Rasterplatte abgeschnitten und mit zwei selbst gefertigten Winkeln befestigt.

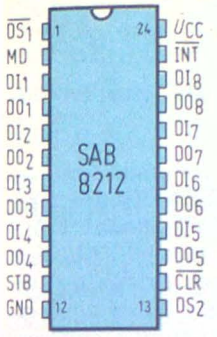
Um die Daten eines Speicherplatzes lesen zu können, oder neue Daten in diesen Speicherplatz einschreiben zu können, muß der Speicherplatz adressiert werden. Dies geschieht über den Adreßbus. Zusätzlich muß der Speicher ein Signal bekommen, ob sein Inhalt ausgelesen werden soll oder ob neue Daten in diesen Platz eingeschrieben werden sollen. Im Falle eines Festwertspeichers (ROM) ist natürlich nur das Lesesignal erforderlich. Will man von außerhalb des Mikrocomputersystems zum Speicher zugreifen, so muß für diese Zeit der Mikroprozessor außer

Tabelle der Bauelemente

4	LED-Anzeigen HA 1143 r	(Siemens AG)
2	LED-Anzeigen HA 1143 g	(Siemens AG)
4	LED-Zeilen LD 464	(Siemens AG)
2	LED-Zeilen LD 474	(Siemens AG)
4	Hex-Inverter-Treiberstufen HEF 4049	(Valvo, Fairchild, National)
6	Hexadezimal-7-Segment-Dekoder-Treiber 9368	(Fairchild)
4	Sechsfacher Inverter 7404 (FLH 211)	(Siemens AG)
1	Monostabile Kippstufe 74123 (FLK 121)	(Siemens AG)
1	LED LD 481	(Siemens AG)
1	Transistor BC 238	(Siemens AG)
1	Diode BA 127	(Siemens AG)
1	Diode AA 118	(Siemens AG)
1	Styroflex-Kondensator, 220 pF, 25 V–, B 31310 – A 3221 – H	(Siemens AG)
1	Keramik-Kondensator, 0,1 µF, 63 V–, B37449–A6104–S2	(Siemens AG)
1	Tantal-Elko 68 µF, 16 V	
1	Widerstand, 820 Ω	
24	Widerstände 1,2 kΩ	
1	Widerstand 4,7 kΩ	
1	Widerstand 10 kΩ	
1	Widerstand 33 kΩ	
1	Einpoliger Schalter	
1	Einpolige Taste	
1	Steckerleiste DIN 41612 mit 64 Anschlüssen, Bauform C	(Siemens AG)
	Zwischenfassungen für die IS und LED-Anzeigen nach Bedarf	
1	Satz Teile für Fädeltechnik:	(Siemens AG)

Die mit (Siemens AG) bezeichneten Bauteile können über den Siemens Bauteile Service, Postfach 146, 8510 Fürth, bezogen werden. Ein Katalog der dort erhältlichen Bauteile wird auf Anforderung (Tel. 09 11/3 00 11) kostenlos zugesandt.

Anschlußbelegung



Anschlußbezeichnungen

DI_1... DI_8	Daten-Eingänge
DO_1... DO_8	Daten-Ausgänge
$\overline{DS_1}$... DS_2	Baustein-Auswahl
MD	Betriebsart
STB	Übernahme
$\overline{INT}$	Unterbrechung (low aktiv)
$\overline{CLR}$	Löschen (low aktiv)
U_{CC}	Versorgungs-spannung (+5V)
GND	Masse

Bild 14. Anschlußanordnung des 8-bit-Eingabe/Ausgabe-Bausteines SAB 8212 (Ansicht von oben)

Funktion gesetzt werden, damit nicht gleichzeitig vom Mikroprozessor Steuerungen vorgenommen und Adressen ausgegeben werden. Der Mikroprozessor läßt sich über den Anschluß „HOLD" zeitweilig außer Funktion setzen. Seine Anschlüsse zu dem Adreßbus und dem Datenbus werden dabei in den hochohmigen Zustand versetzt.

Modul für direkten Speicherzugriff

Bild 13 zeigt die Schaltung des Moduls für den direkten Speicherzugriff. Mit den Adreßschaltern A_0 bis A_{15} wird die gewünschte Adresse eingestellt. Die Schalter legen an die Bausteine 8212 entweder in der Stellung nach unten ein L-Signal, in der Stellung nach oben ein H-Signal. Die 16-stellige Adresse setzt sich aus vier Sedezimalzahlen zusammen. Wird der Schalter „direkter Speicherzugriff" geschlossen, so werden die Bausteine 8212 über die Freigabeleitung $\overline{DS_1}$ aktiviert. Gleichzeitig gelangt über die Leitung 17 a ($\overline{HOLD}$) ein L-Signal auf den Steuerbus, der den Mikroprozessor abschaltet. Er signalisiert die Hochohmigkeit seiner Adreß- und Datenbusse mit dem HLDA-Signal (Hold Acknowledge), welches über die DS 2 Freigabe-Eingänge die oberen beiden Bausteine 8212 freigibt. Ferner gelangt über die Gatter G 1 und G 2 auf die Leitung 8 a ($\overline{MEM\ R}$) L-Signal, welches dem Speicher sagt, daß er gelesen wird. Die im Speicher befindlichen Daten werden vom Speicher auf den Datenbus gegeben und können beispielsweise mit dem Anzeigemodul angezeigt werden. Der Anzeigemodul ist ebenfalls an den Adreßbus angeschlossen, so daß auch die dazugehörige Adresse auf dem Anzeigemodul erscheint. Auf diese Art können also sowohl Festwertspeicher (ROM) als auch Schreib/Lesespeicher (RAM) gelesen werden.

Sollen jetzt neue Daten in einen Schreib/Lesespeicher eingeschrieben werden, so sind die Daten mit den Datenschaltern D_0 bis D_7 einzustellen. Die dazugehörigen L- bzw. H-Pegel gelangen an die Eingänge des unteren Bausteines 8212 (*Bild 13*). Beim Drücken der Taste „Daten einschreiben" wird über das Gatter G 3 ein H-Signal auf den Freigabeanschluß DS_2 gegeben; der Baustein 8212 schaltet damit die an den Schaltern eingestellten Daten auf den Datenbus durch. Gleichzeitig bekommt die Leitung 10 a ($\overline{MEM\ W}$) über das Gatter G 4 L-Signal. Der Speicher wird auf „Einschreiben" geschaltet. Da der untere Anschluß des Gatters G 2 von der Taste her L-Signal erhält, wird das vorher am Anschluß 8 a stehende Signal ($\overline{MEM\ R}$) gelöscht.

Die Klemmen STB der Bausteine 8212 sind über eine Drahtbrücke (gestrichelt gezeichnet) mit +5 V verbunden. Damit erfolgt in den Bausteinen beim Umlegen der Adreß- und Datenschalter eine sofortige Datenübernahme. Möchte man für die Datenübernahme eine Taste vorsehen, so ist statt der Drahtbrücke die rechts daneben stehende Hilfsschaltung einzusetzen. *Bild 15* zeigt den aufgebauten Modul in Fädeltechnik auf einer Lochrasterplatine. *(Fortsetzung folgt)*

Tabelle der Bauelemente

3 Eingabe/Ausgabe-bausteine 8212, Bild 14 (Siemens AG)
1 Baustein, 4 NAND-Glieder mit je zwei Eingängen und offenem Kollektor 7401 (FLH 201) (*Bild 16*) (Siemens AG)

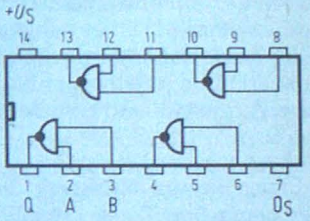


Bild 16. Anschlußanordnung des Bausteines Vier NAND-Glieder mit je zwei Eingängen und offenem Kollektor 7401 (FLH 201) (Ansicht von oben)

2 Dioden AA 118 (Siemens AG)
2 Kondensatoren 0,1 µF, 63 V– B 37443 – B 0104 – S 3 (Siemens AG)
1 Widerstand 100 Ω
5 Widerstände 4,7 kΩ
24 einpol. Umschalter
1 einpoliger Schalter
1 einpolige Arbeitstaste
bei Bedarf:
1 einpolige Ruhetaste
1 Widerstand 4,7 kΩ
1 Satz Teile für Fädeltechnik (siehe Folge 2) (Siemens AG)

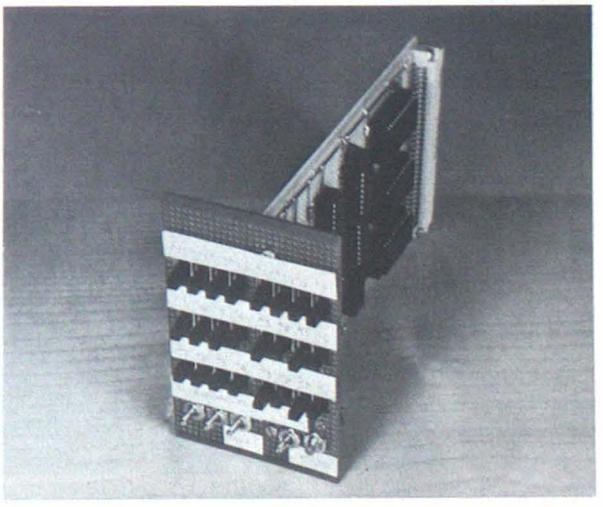


Bild 15. Foto des aufgebauten Moduls für direkten Speicherzugriff mit Zusatzschaltung zur Anzeige der Daten einer vorgewählten Adresse im Betrieb

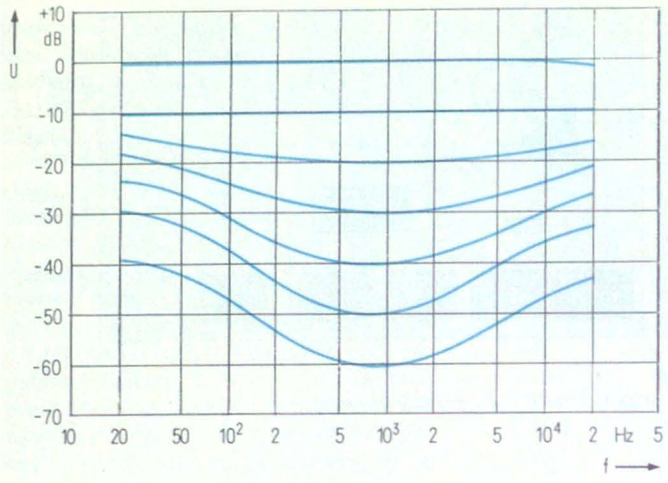


Bild 5. Wirkung der physiologischen Lautstärkeeinstellung

über Mittelstellung erreichen, ein Bereich, der erstens völlig ausreicht und zweitens die Gesamtlautstärke im Raum beim Verdrehen des Balancestellers unverändert aufrechterhält.

Klangeinsteller

Während beim üblichen passiven Klangeinsteller (Kuhschwanzregler) im allgemeinen schon kleine Toleranzen der frequenzbestimmenden Bauteile (wegen der damit verbundenen Veränderung der Zeitkonstanten) zu erheblicher Welligkeit des Frequenzganges führen können, ist dieser Einfluß hier ausgeschlossen. Das wird beim Tiefensteller dadurch erreicht, daß die Kondensatoren für die Anhebung bzw. Absenkung der Tiefen in der Mittelstellung durch eine schalterähnliche Zusatzfunktion kurzgeschlossen werden. Beim Höhensteller genügt es dagegen, den Schleifer des Potentiometers durch geeignete konstruktive Ausbildung in der Mittelstellung abzuheben, um die entsprechenden frequenzabhängigen Schaltungsteile abzutrennen. Um das definierte Einstellen der Klangfarben und besonders der Mittelstellung zu erleichtern, sind die Steller mit je 11 Raststellungen ausgerüstet.

Rausch- und Rumpelfilter

Die geforderte Verzerrungsarmut machte beim Endverstärker eine Erhöhung des Gegenkopplungsgrades notwendig. Die dadurch eingetretene Verringerung der Verstärkung wird durch den 2. Zwischenverstärker ausgeglichen, in dessen Ausgang sich das Rauschfilter befindet.

Das Rumpelfilter ist als aktives Filter ausgebildet. Es ist jedoch unmittelbar vor dem Endverstärker angeordnet und an dessen nichtinvertierenden Eingang angeschlossen, während die rückkoppelnden Komponenten des Filters mit dem invertierenden Eingang des Endverstärkers verbunden sind. Da die Eingänge praktisch gleichspannungsfrei sind, entstehen beim Ein- und Ausschalten des Rumpelfilters sowie durch den hier ebenfalls angeschlossenen elektronischen Stummschalter (Muting) keine Störgeräusche.

Der Endverstärker

Auch beim Endverstärker wurde die Technik des Spitzengerätes TRX 2000 mit symmetrischer Stromversorgung, kondensatorlosen Lautsprecherausgängen und Differenzverstärkereingängen übernommen. Die sich dadurch ergebenden geringen Verzerrungen ($k_{nenn} \leqq 0{,}1$ %) und der niedrige Ausgangsscheinwiderstand ($R_i \leqq 0{,}25\ \Omega$) lassen sich durchaus mit den technischen Daten des Spitzengerätes vergleichen. Ermöglicht wurde dies durch die neue integrierte Schaltung LM 391, die alle Komponenten zur Ansteuerung der Treiber- und Endtransistoren enthält (Eingangsdifferenzverstärker, Vortreiber mit kollektorseitiger Konstantstromquelle, Transistor zur Ruhestromstabilisierung). Die ebenfalls in der LM 391 untergebrachte elektronische Sicherung arbeitet als Kompensationsschaltung mit rückläufiger Stromkennlinie (fold back).

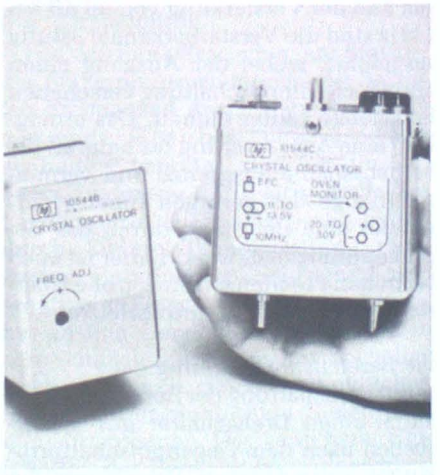
Neben der Peripheriebeschaltung für diese IS, die hauptsächlich für die Festlegung der Arbeitspunkte, die Gegenkopplung und die elektronische Sicherung gebraucht wird, werden nur noch die Treiber- und Endstufenpaare benötigt. Da neben Lautsprechern mit einer Impedanz von 8 Ω auch solche mit 4 Ω – bei nur wenig verminderten technischen Daten – verwendbar sein sollen, sind die Endtransistoren und die Kühlkörper sehr großzügig ausgelegt. Zusammen mit den weiteren Maßnahmen, wie der erwähnten elektronischen Sicherung, Thermoschaltern auf dem Kühlkörper und im Netztransformator sowie der Abschaltautomatik für den Fall des Auftretens von Fehlströmen im Lautsprecherkreis, ist damit Vorsorge gegen alle möglichen Überlastungszustände getroffen.

Die an den Lautsprecherbuchsen bereitgestellte Leistung beträgt 2 x 50 W an 8 Ω für k ≦ 0,1 % und 2 x 70 W an 4 Ω für k ≦ 0,1 %. Für alle Betriebsarten ist der auf 50 mW bezogene Fremdspannungsabstand nach DIN (mit Spitzenbewertung) ≧ 60 dB.

Die Stabilisierung der Spannungen, mit denen die Vorstufen und die Hf-Teile versorgt werden, geschieht mit Hilfe neuer einstellbarer Stabilisierungs-IS, wodurch ein stabiles Arbeiten der einzelnen Komponenten bei optimalen Betriebsbedingungen gewährleistet ist. ■

10-MHz-Quarzoszillator

Zwei neue Quarzoszillatoren von Hewlett-Packard sind für Anwendungen vorgesehen, bei denen es besonders auf geringe Alterungsraten und hohe Temperaturstabilität ankommt. Sie enthalten Thermostaten mit einer Aufwärmzeit von etwa 20 Minuten; das Phasenrauschen – ein Qualitätsmaßstab für Frequenznormale – liegt in 1 kHz Abstand vom Träger bei –150 dB. Die Alterung wird mit $5 \cdot 10^{-10}$ pro Tag angegeben.



Mit einem Schraubendreher kann ein Feinabgleich der Ausgangsfrequenz vorgenommen werden. Der Typ HP 10544 B verfügt über einen Stecker für den Einbau in gedruckte Schaltungen, während sein Artgenosse HP 10544 C einen separaten Hf-Anschluß und gefilterte Zuleitungen besitzt. Die Abmessungen sind bei beiden Ausführungen gleich (*Bild*): 72 mm x 52 mm x 62 mm; das Gewicht beträgt 310 p. Fe.

Horst Pelka

Mit der in dieser Folge veröffentlichten Zusatzschaltung lassen sich sämtliche Speicherplätze auch während des Betriebes anzeigen. Weiter wird mit der Beschreibung des Netzteiles begonnen.

Praxis mit Mikroprozessoren

3. Teil

Zusatzschaltung zur Anzeige der Daten von einer vorgewählten Adresse

Diese Schaltung (*Bild 17*) gestattet zusammen mit dem Anzeigemodul, die Daten einer vorgewählten Adresse *während des Betriebes des Mikroprozessors* anzuzeigen. Die Schaltung kann entweder separat auf einer Europa-Karte aufgebaut werden oder auf dem Modul für direkten Speicherzugriff untergebracht werden. Beim Aufbau auf eine gemeinsame Karte spart man die 16 Adressenschalter.

Die mit dem Adressenschalter eingestellte Adresse wird mit vier 4-Bit-Komparatoren mit der gerade auf dem Adreßbus befindlichen Adresse verglichen. Um den Adreßbus nicht zu belasten, sind Komparatoren in CMOS-Technik gewählt worden (Bild 17).

Besteht Adressengleichheit zwischen den Adressenschaltern und dem Adreßbus, geht der Ausgang (Anschluß 3) des unteren Komparators auf H. Der nachgeschaltete Inverter 7401 invertiert dieses H-Signal und gibt auf den Anschluß 17 des Systembusses ein L-Signal. Dieses L-Signal bewirkt die Freigabe der Eingangsspeicher in dem auf dem Anzeigemodul befindlichen Hexadezimal/7-Segment-Dekodern.

Mit einem Umschalter können zwei Betriebsarten gewählt werden. In der Stellung „Speicher" werden alle 16 Bit des Adreßbusses für den Vergleich herangezogen. Ferner muß entweder ein Signal (MEM R) oder (MEM W) vorhanden sein. In der Stellung EA werden nur 8 Bit des Adreßbusses zum Vergleich herangezogen. Mit Einzelschaltern kann zusätzlich noch gewählt werden, ob ein Vergleich nur bei Eingabe- oder bei Ausgabedaten oder bei beiden stattfindet.

Nachdem der Anzeigemodul und der Modul für direkten Speicherzugriff fertiggestellt sind, können beide geprüft

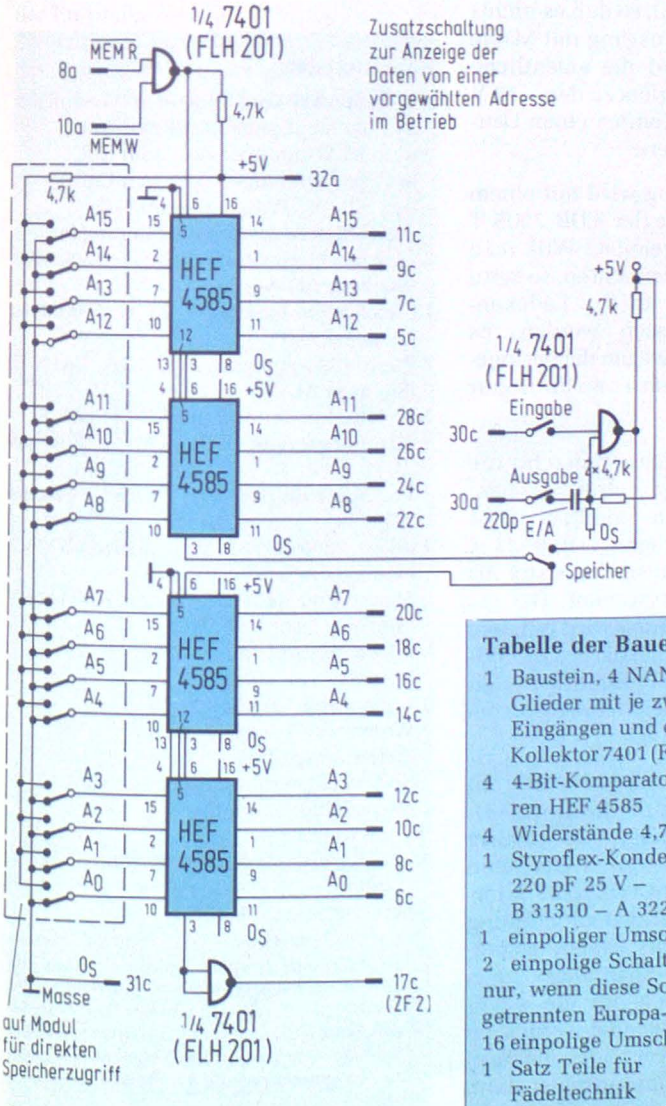
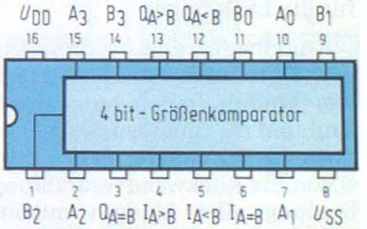


Bild 17. Zusatzschaltung zur Anzeige der Daten einer vorgewählten Adresse im Betrieb

Bild 18. ▶ Anschlußanordnung der 4-Bit-Komparatoren HEF 4585 (Ansicht von oben)



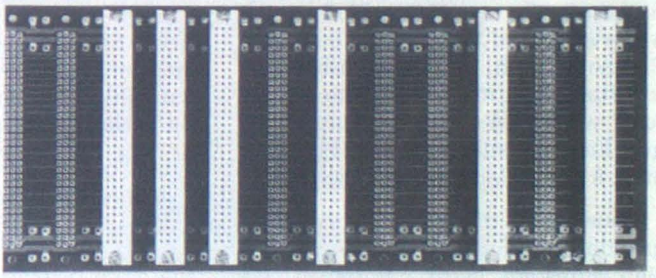


Bild 19. Platine für die Rückwandverdrahtung

Tabelle der Bauelemente

1 Baustein, 4 NAND-Glieder mit je zwei Eingängen und offenem Kollektor 7401 (FLH 201) (Siemens AG)
4 4-Bit-Komparatoren HEF 4585 (Valvo, NSC)
4 Widerstände 4,7 kΩ
1 Styroflex-Kondensator 220 pF 25 V – B 31310 – A 3221–H (Siemens AG)
1 einpoliger Umschalter
2 einpolige Schalter
nur, wenn diese Schaltung auf einer getrennten Europa-Karte aufgebaut wird.
16 einpolige Umschalter
1 Satz Teile für Fädeltechnik (Siemens AG)
1 Widerstand 4,7 kΩ

werden. Der Schalter „direkter Speicherzugriff" wird geschlossen. Damit bekommt über die Diode AA 118 die Steuerbusleitung 17 c (ZF 2) L-Signal. Die im Anzeigemodul befindlichen Hexadezimal-7Segment-Decoder erhalten dieses L-Signal an ihren Anschlüssen 3. Es werden alle auf den Adreß- und Datenbussen befindlichen Daten übernommen. Die am Modul für direkten Speicherzugriff mit den Adressenschaltern eingestellten Adressen müssen an den vier dafür zuständigen LED-Displays im Anzeigemodul erscheinen. Gleichzeitig geben die LED-Zeilen die betreffenden Daten bitweise wieder.

Die mit den Datenschaltern eingestellten Daten erscheinen erst auf dem

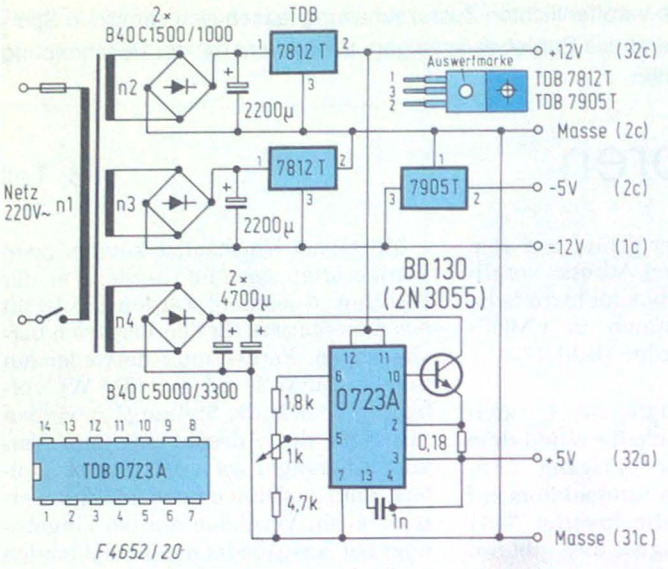


Bild 20. Schaltung des Netzteils

Datenbus, wenn die Taste „Daten einschreiben" gedrückt ist. Jetzt müssen auch die beiden Datenanzeigen im Anzeigemodul die an den Schaltern eingestellten Daten anzeigen,das gleiche gilt für die LED-Zeilen.

Grundbedingung für diese Überprüfung ist selbstverständlich, daß die beiden Moduln miteinander verbunden sind und die Speisespannung +5V anliegt. Es ist daher zweckmäßig, jetzt schon die Rückwandverdrahtung (Verbindung aller Moduln miteinander) aufzubauen. Für die Rückwandverdrahtung gibt es geätzte Platinen als Meterware. *Bild 19* zeigt ein Foto dieser Platine. Obwohl die erste Ausbaustufe nur fünf Moduln vorsieht (ohne Netzteil), wird vom Verfasser vorgeschlagen, für den weiteren Ausbau Plätze für etwa 8 bis 10 Moduln vorzusehen.

Diejenigen, die aus Sparsamkeitsgründen das System ohne Steckerleisten aufbauen, wird empfohlen, die Moduln mit Winkeln an entsprechende Rasterplatten anzuschrauben und mittels Fädeltechnik die Moduln untereinander zu verbinden. Die Masseleitung und die Speisespannungsleitungen müssen mit dickem Draht (> 1,5 mm∅) miteinander verbunden sein.

Netzteil

Wer kein Labornetzgerät mit einer Speisespannung von 5 V bei einem Strom bis zu etwa 1,5 bis 2 A zur Verfügung hat, sollte den Bau des Netzteiles vorziehen.

Für die ersten Ausbaustufen genügt in der Regel ein konventionelles Netzteil mit 50-Hz-Netztrafo (*Bild 20*). Auf der Sekundärseite des Trafos befinden sich Wicklungen mit 8 V für die 5-V-Speisespannung und 2x 15 V~ für die 12-V-Speisespannungen.

Für die + 12V wird der Positivspannungsregler TDB 7812 T verwendet, ebenso für die 12-V-Speisespannung der gleiche Typ des Positivspannungsreglers. Der Innenwiderstand des TDB 7812 T beträgt 18 mΩ, so daß es nichts ausmacht, daß der Ausgang mit Masse verbunden wird, und die eigentliche Masseleitung des Reglers zu den – 12 V führt. Beide Regler können einen Dauerstrom von 1 A liefern.

Die – 5-V-Spannung wird mit einem Negativ-Spannungsregler TDB 7905 T aus den – 12 V abgeleitet. Will man ganz auf die – 12 V verzichten, so kann dieser Regler direkt an den Ladekondensator angeschlossen werden. Es empfiehlt sich dann wegen der geringeren Verlustleistung am Trafo für n_3 nur 60 Wdg. vorzusehen.

Um für weitere Ausbaustufen für die +5V-Speisespannung eine größere Stromergiebigkeit zu erhalten, wird dort der Spannungsregler TDB 0723 A und ein nachgeschalteter Transistor BD 130 oder 2N3055 verwendet. Die genaue Ausgangsspannung wird mit dem 1-kΩ-Potentiometer justiert. Der Widerstand 0,18Ω vom Anschluß 2 nach Anschluß 3 der IS ergibt eine Strombegrenzung bei etwa 3 A.

Um die Verlustleistung des BD 130 gering zu halten, wird die unstabilisierte Spannung möglichst niedrig (hier etwa 8 V) dimensioniert. Die Daten des Spannungsreglers TDB 0723 A fordern jedoch eine Eingangsspannung > 9,5 V. Daher wird der Spannungsregler an die +12-V-Spannung angeschlossen. Soll jedoch nur die +5-V-Spannung allein verwendet werden, so muß die Spannung am Ladekondensator (an den der Spannungsregler dann mit angeschlossen wird) auf mindestens 10 V erhöht werden. Die Wicklung n_4 bekommt dann 60 Wdg. und das Kühlblech für den BD 130 ist entsprechend größer auszulegen.

Mit Ausnahme des Netztrafos können sämtliche Bauteile auf einer Raster-Platine in Europaformat untergebracht werden. Diese Platine kann entweder wie ein Modul in die Rückwandverdrahtung eingesteckt werden. Dann sind die drei Wechselspannungen am besten über eine Stiftleiste getrennt zuzuführen. Eine andere Möglichkeit besteht in der Unterbringung dieser Netzteilplatine im Gehäuse neben oder hinter der Rückwandverdrahtung zusammen mit dem Netztrafo.

Wollen Sie das Netzteil als sogenanntes Schaltnetzteil aufbauen, so finden Sie in [4] einen dementsprechenden Schaltungsvorschlag. Dieses Netzteil hat auf der 5-V-Speisespannung eine Stromergiebigkeit von 5 A.

Literatur

[4] Siemens-Schaltbeispiele, Ausgabe 1977/78, S.151-153. Bestell-Nr. B 1741

Bauteileliste:

1 Netztrafo M 74 mit folgenden Wicklungen:
n_1 1237 Windungen 0,27 mm CuL
n_2 85 Windungen 0,6 mm CuL
n_3 85 Windungen 0,37 mm CuL
n_4 45 (60) Wdg. 1,6 mm CuL
ausreichende Zwischenisolierungen
2 Gleichrichterbrücken B 40 C 1500/1000 (Siemens AG)*
1 Gleichrichterbrücke B 40 C 5000/3300 (Siemens AG)*
2 Positiv-Spannungsregler TDB 7812 T (Siemens AG)*
1 Negativ-Spannungsregler TDB 7905 T
1 Spannungsregler TDB 0723 A (Siemens AG)*
1 Transistor BD 130 oder 2N3055 (Siemens AG)*
2 Elkos 2200 µF/25 V B 41010-C5228-T (Siemens AG)*
2 Elkos 4700 µF/16 V B 41010-B4478-T (Siemens AG)*
1 Einstellregler 1 kΩ
1 Widerstand 0,18 Ω/2 W
1 Widerstand 1,8 kΩ/0,25 W
1 Widerstand 4,7 kΩ/0,25 W
1 Sicherungshalter
1 Schmelzsicherung mittelträge 0,2A
1 Netzschalter, einpolig
Bedarfsweise:
1 Rückwandverdrahtung B 88080 – A 3 – S 400**
für SMP-System S 400

* Die Bauteile (außer Transformator) können über Siemens Bauteile Service Postfach 146, 8510 Fürth bezogen werden. Ein Katalog der erhältlichen Teile wird von dort bei Anforderung kostenlos zugesandt; Tel. (0911) 30011.

** Anfragen über Bezugsmöglichkeit an Ihre nächste Zweigniederlassung der Siemens AG, Abt. VB.

(Fortsetzung folgt)

Horst Pelka

In den vorhergehenden Folgen wurde der Anzeigemodul, der Modul für direkten Speicherzugriff und das Netzgerät beschrieben. Die am Modul für direkten Speicherzugriff eingestellten Adressen und Daten konnten am Anzeigenmodul sichtbar gemacht werden.

Praxis mit Mikroprozessoren

4. Teil

Prüfung der Moduln

Das Netzteil wird in Betrieb genommen. Die beiden Spannungen –5 V und + 12 V werden mit 0,5 A, die Versorgungsspannung + 5 V mit 2,5 A belastet. Die Spannungen +12 V, – 12 V und – 5 V müssen eine Toleranz von ± 5 % haben. Mit dem 1-kΩ-Potentiometer wird die + 5-V-Versorgungsspannung auf genau 5 V ± 0,25 eingestellt. Jetzt wird der Modul für direkten Speicherzugriff eingesteckt bzw. angeschlossen. Die Stromentnahme bei den 5 V beträgt ca. 300 mA. Als nächstes wird der Anzeigemodul eingesteckt bzw. angeschlossen. Die Gesamtstromaufnahme beträgt jetzt 1,3A. Nun prüft man die richtige Beschaltung der Adreß- und Datenschalter bzw. Adreß- und Datenanzeigen folgendenmaßen:

Jeweils ein Viererbündel, dies sind die Adreßleitungen A 0...A 3, A 4...A 7, A 8...A 11, A 12...A 15 und die Datenleitungen D 0...D 3, D 4...D 7, wird entsprechend der Code nach Bild 4 durchgezählt. Die an den Schaltern eingestellten Pegel H lassen die zugehörigen LED aufleuchten. Gleichzeitig erscheint an den Displays die Darstellung der 4 Bit in Hexcode. Zu beachten ist, daß die Daten des Datenbusses nur angezeigt werden, wenn die Taste „Daten einschreiben" gedrückt ist. Evtl. überbrückt man während der Überprüfung die Taste mit einer Laborschnur.

Es ist darauf zu achten, daß sich jeweils nur die Ziffer in der einen Vierer-Bit-Gruppe verändert. Eventuelle Schaltfehler ergeben sich aus der entsprechenden Fehlanzeige.

Sollten trotz richtiger Beschaltung die Displays nicht die richtigen Adressen und Daten anzeigen, so müssen mit einem hochohmigen Meßinstrument ($R_i > 50\,000\ \Omega/V$) die einzelnen Pegel nachkontrolliert werden. Alle H-Pegel sollen > 2,4 V sein, alle L-Pegel < 0,4 V. Man nimmt sich am besten, beginnend bei A_0 zuerst alle Adreßleitungen vor und dann die Datenleitungen. Adreß- und Datenschalter werden einmal auf L-Pegel, einmal auf H-Pegel gestellt. Besonderes Augenmerk sollte den Hexpuffern 4049 geschenkt werden. Es kommt schon mal vor, daß billige Sonderangebote nicht die Datenblattwerte einhalten.

Schaltung zur Datensperre während des Ein- und Ausschaltens

Auf der Netzteilplatine sollten Sie noch etwas Platz für eine Hilfsschaltung zur Sperre von Speicherbausteinen und anderen Funktionsgruppen während des Ein- und Ausschaltvorganges (*Bild 21*) lassen. Diese Schaltung liefert am Ausgang H-Pegel (ca. 2,5 V), wenn die volle Speisespannung vorhanden ist. Eine solche Schaltung benötigt man u.a. bei batteriegepufferten CMOS-Speichern, weil auf den Steuerbussen, Adreß- und Datenleitungen bei absinkender Spannung nach dem Abschalten undefinierte Signale vorhanden sind, die beispielsweise den Speicher auf „Daten einschreiben" stellen können und so bei der zufällig auf dem Adreßbus sich einstellenden Adresse die gerade auf dem Datenbus stehenden Daten in die Speicherzelle einschreiben.

Die Funktion ist folgende: Wird das Netz abgeschaltet, so sinkt die Spannung am Kondensator 2,2 µF innerhalb von 30 ms unterhalb $^1/_4$ der Normalspannung, und der Ausgang gibt niederohmig an 220 Ω L-Pegel ab, mit dem über den Freigabe-Eingang der Speicher gesperrt werden kann. Ebenso erzeugt der linke Transistor BC 238 B bei Reset-Signal an ZF 1 (H-Pegel) ein L-Signal an ZF 0. Dieses Signal führt über den Anschluß 13c (ZF 0) zu den anderen Moduln. Es sollte erwähnt werden, daß bei den fertigen Rückwandverdrahtungen *sämtliche* Zusatzfunktionen (ZF 0...8) nicht durchverbunden sind, also mit Hand entsprechend beschaltet werden müssen.

Der Abfall der 5-V-Speisespannung unter den Wert 4,75 V (Toleranzgrenze für TTL-Schaltungen) dauert länger. Beim Einschalten des Netzes führt 13c H-Pegel, wenn die Speisespannung etwa 4 V erreicht hat. Dieser Vorgang ist nicht so kritisch, weil beim Einschalten ein automatischer RESET im Mikroprozessormodul wirksam ist.

Da ein Mikroprozessor ohne Programmspeicher nicht funktionsfähig ist, wird als nächstes der Speichermodul beschrieben. Zusammen mit den anderen drei Moduln ist der Leser in der Lage, das Anwenderprogramm in den CMOS-Speicher einzuschreiben. Der CMOS-Speicher ist batteriegepuffert, so daß er auch bei Ausfall der Netzspannung seine Daten behält.

Allgemeines über Halbleiterspeicher

Bei Mikroprozessorsystemen unterscheidet man zwischen Fest(wert)-speicher und Schreib/Lesespeicher. Bei Festspeichern wird der Inhalt einmalig eingeschrieben. Nach diesem Vorgang kann der Inhalt dann nur noch ausgelesen werden. Im englischen Sprachgebrauch werden die Festspeicher mit ROM (Read Only Memory) bezeichnet. Hier wird der Speicherinhalt bereits während des Herstellungsvorganges durch eine Verdrahtungsmaske bestimmt.

Festspeicher, die durch den Anwender selbst programmiert werden können, nennt man PROM (programmierbare ROM). Bei ihnen wird der Inhalt nach Anlegung der richtigen Adresse mit einem Schreibeimpuls, genannt Programmimpuls, eingeschrieben. PROM's können nach dem Einschreiben nicht mehr gelöscht werden.

Bauteileliste

2 Transistoren BC 238 B (Siemens AG)
2 Dioden BA 127 d (Siemens AG)
1 Z-Diode BZX 83 C 6V2 (Siemens AG)
1 MKH-Schichtkondensator 2,2 µF 100 V B 32560-B1225-J (Siemens AG)
1 Widerstand 150 Ω, 0,25 W
1 Widerstand 220 Ω, 0,25 W
1 Widerstand 1 kΩ, 0,25 W
3 Widerstände 10 kΩ, 0,25 W

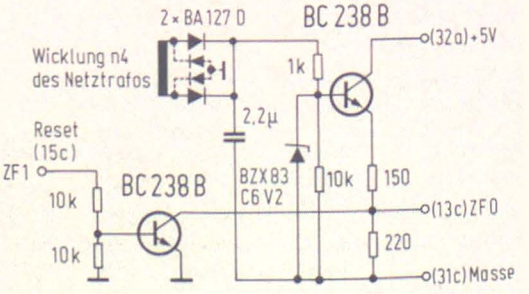


Bild 21. Hilfsschaltung zur Datensperre während des Ein- und Ausschaltens. Die gestrichelt angeschlossenen Dioden befinden sich im Brückengleichrichter B 40 C 5000/3300 (Bild 20 der letzten Folge)

Will man den Inhalt auch wieder löschen können, muß man REPROMs einsetzen. Dies sind löschbare und elektrisch neu programmierbare Festspeicher. Das Einschreiben des Speicherinhaltes geschieht wie bei einem PROM. Der Speicherinhalt wird gelöscht, wenn der Halbleiterchip starken ultravioletten Strahlen mit einer Wellenlänge von 2537 Å ausgesetzt wird. Als Dosis genügen 6...10 Ws/cm². Als Quelle für das ultraviolette Licht kann z. B. ein Entkeimungsstrahler der Firma Osram Typ HNS verwendet werden. Er löscht die REPROM's in etwa 10...30 min bei 2...3 cm Abstand.

Ferner werden in Mikroprozessorsystemen Schreib/Lesespeicher verwendet, im englischen Sprachgebrauch RAM (Random Access Memory). Bei dieser Art von Speichern kann man jederzeit durch elektrische Umschaltung den Speicherinhalt verändern, d.h. der Speicher hat zwei Arbeitsstellungen,

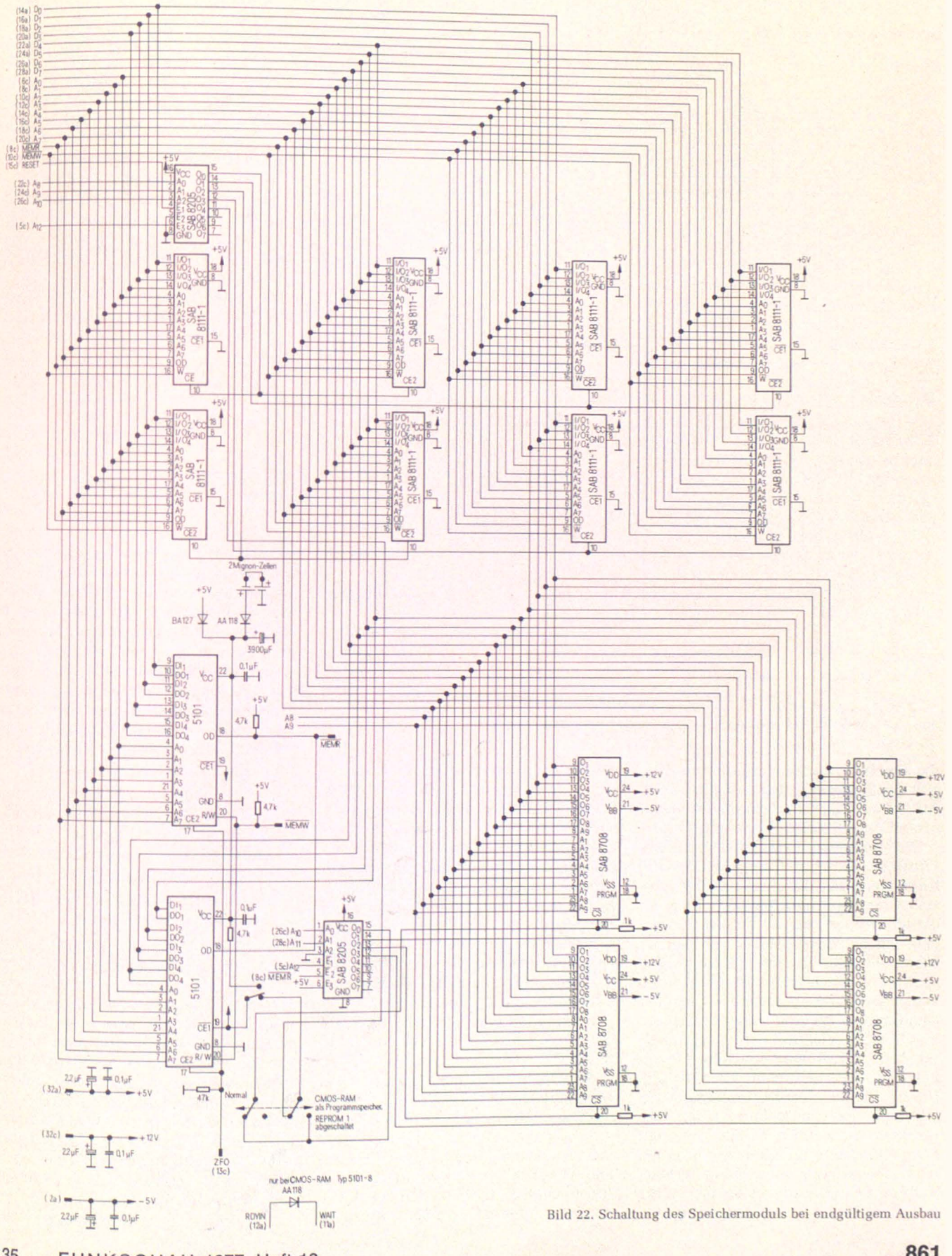


Bild 22. Schaltung des Speichermoduls bei endgültigem Ausbau

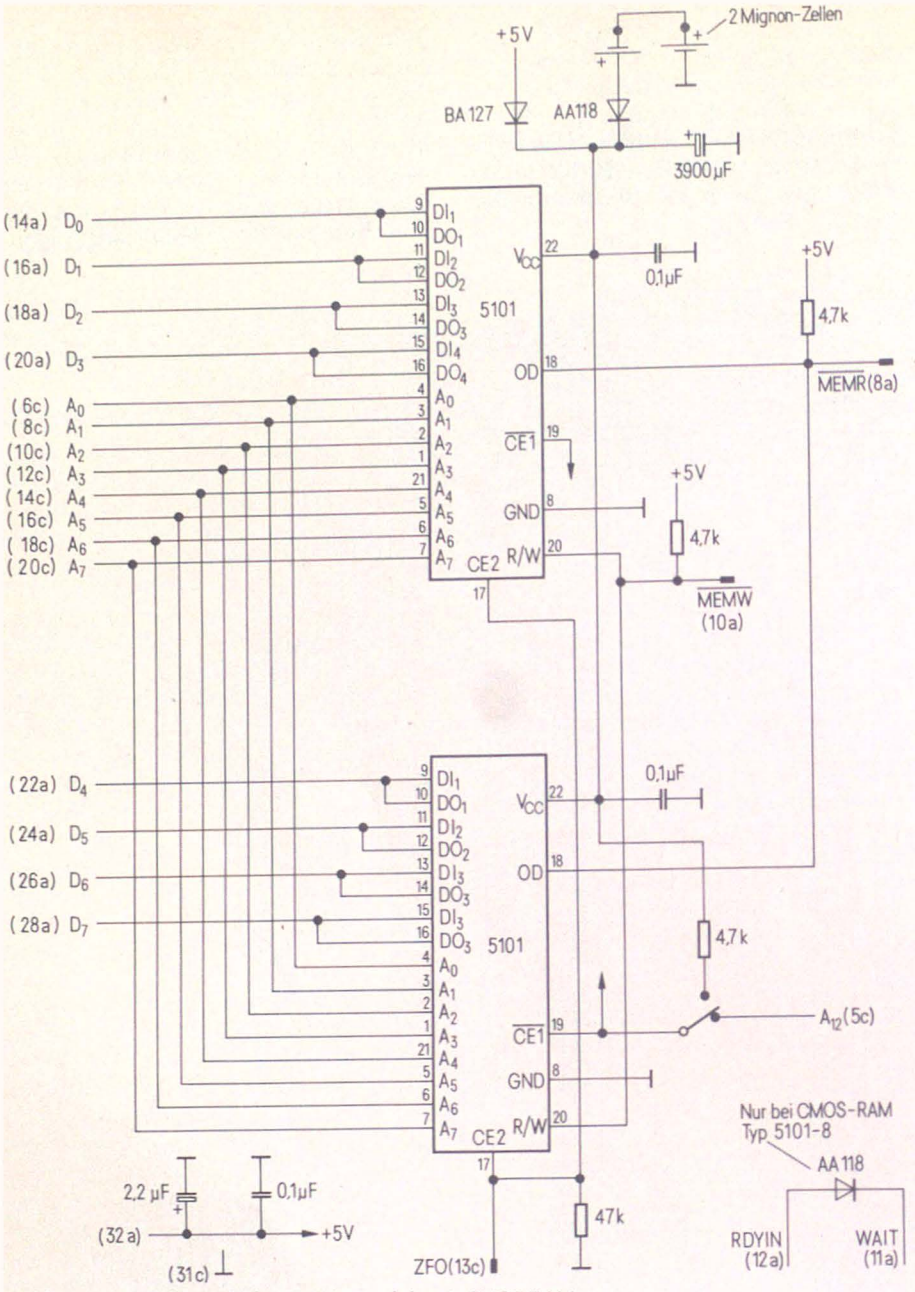


Bild 23. Erste Ausbaustufe des Speichermoduls mit CMOS-RAMs

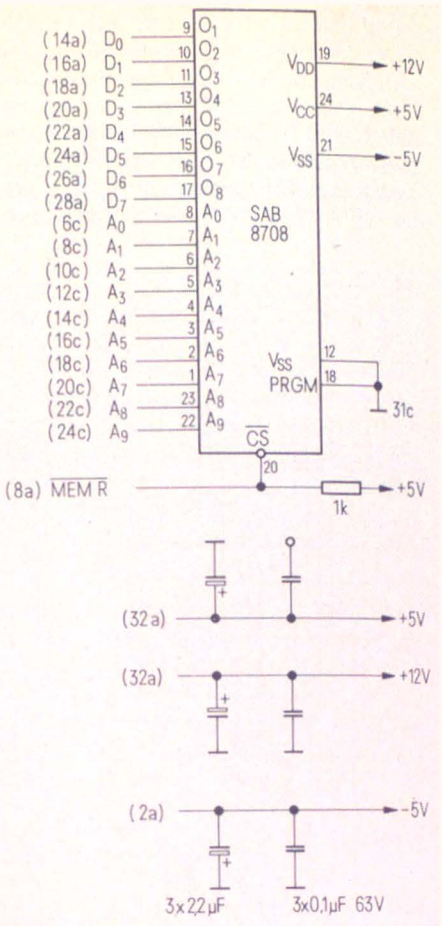


Bild 24. Erste Ausbaustufe des Speichermoduls mit REPROM

die eine heißt „Lesen", in diesem Falle gibt der Speicher Daten aus, die zweite Arbeitsstellung ist „Schreiben", dabei nimmt der Speicher Daten auf.

Die einzelnen Zellen bestehen aus Flipflops. Da beim Abschalten der Betriebsspannung der ursprüngliche Zustand des Flipflops nicht mehr erhalten bleibt, verliert der Schreib/Lesespeicher im Gegensatz zum Festspeicher beim Abschalten der Versorgungsspannung seinen Inhalt. Dies kann man verhindern, indem man am Speicher ständig eine kleine Versorgungsspannung beläßt, die groß genug sein muß, um den Ursprungszustand der einzelnen Flipflops aufrecht zu erhalten. Diese kleine Versorgungsspannung entnimmt man entweder kleinen Monozellen oder NiCd-Zellen. Um den Stromverbrauch möglichst gering zu halten, verwendet man für batteriegepufferte Speicher am besten Typen, die in CMOS-Technik aufgebaut sind.

Um die entsprechenden Speicherzellen auswählen zu können, hat jeder Speicher Adresseneingänge, und zwar soviel Adressen, wie der Speicher Speicherplätze hat. Ob die Anzahl der Speicherplätze mit der Anzahl der Adressen korrespondiert, kommt auf die Organisation des Speichers an. Nehmen wir als Beispiel das REPROM SAB 8708, welches 8192 Bit umfaßt. Die Organisation des Speichers ist in 1024 x 8 Bit, d. h. mit einer Adresse werden immer 8 Bit auf einmal angesprochen. Folglich brauchen wir für diesen Speicher nur 1024 Adressen, die mit einem 10-Bit-Adreßbus bedient werden können.

Als weiteres Beispiel die Schreib/Lesespeicher SAB 8111 oder die CMOS-Speicher 5101, welche jeweils 1024 Bit umfassen. Diese beiden Speicher sind jedoch 256 x 4 Bit organisiert, so daß 256 Adressen von außen angesprochen werden müssen. Dies kann man wiederum mit einem Adreßbus von 8 Bit bewerkstelligen. Um eine Datenbreite von 8 Bit, wie sie beim Mikroprozessorsystem 8080 üblich ist, mit solchen Schreib/Lesespeichern abdecken zu können, braucht man für 256 Adressen jeweils zwei Speicher SAB 8111 oder zwei CMOS-Speicher 5101. Die Adressen-Eingänge werden dabei parallelgeschaltet. Der eine Speicher wird für die vier niederwertigen Daten-Bits, der zweite Speicher für die vier höherwertigen Daten-Bits angeschlossen.

Bild 22 zeigt die Gesamtschaltung des auf einer Platine mit Europaformat aufgebauten Speichermoduls. Für den endgültigen Ausbau sind vorzusehen:

4 Stck. REPROM 8708
8 Stck. RAM-Speicher SAB 8111
2 CMOS-RAM-Speicher 5101.

Ferner befinden sich auf dieser Platine noch Adressen-Decoder. Für die ersten „Gehversuche" unseres Mikroprozessors brauchen wir aber nur entweder zwei CMOS-RAMs, wenn wir das Anwendungsprogramm mit unserem Modul für direkten Speicherzugriff selbst in Maschinensprache einschreiben wollen, oder ein REPROM SAB 8708, wenn wir die Möglichkeit haben, dieses auf einem Entwicklungsplatz zu machen. Die erste Ausbaustufe des Speichermoduls vereinfacht sich dann damit zu *Bild 23* oder *Bild 24*.

(Fortsetzung folgt)

Horst Pelka

Im vorliegenden 5. Teil beschreibt der Autor die verschiedenen Speicher sowie das Lesen und Einschreiben in einen Speicher.

Praxis mit Mikroprozessoren

5. Teil

Speicheradressenaufteilung des Speichermoduls

Bild 25 zeigt die Adressenaufteilung des vollbestückten Speichermoduls. Das erste REPROM nimmt die Adressen 0000 bis 03 FF ein, dies sind dezimal ausgedrückt 1024 Stellen. Dann sind weitere Plätze für drei REPROMs mit je 1024 Adressen vorgesehen. 1024 Adressen nennt man auch 1 K. Wir hätten dann also 4 K ROM, denn für die Funktion des Mikroprozessors ist es egal, ob der Festspeicher als ROM, PROM oder REPROM ausgeführt ist.

Ab der Adresse 1000 beginnen die Schreib/Lesespeicher. Hier ist die Bestückung von 1 K RAM vorgesehen. Das CMOS-RAM bekommt im Normalfall die Adressen 1400 bis 14FF zugeteilt. Nur auf diesen Adressen lassen sich in Schalterstellung „Normal" Daten einschreiben.

Mit einem Umschalter kann das CMOS-RAM aber auch auf die Adresse 0000 bis 00FF umgeschaltet werden und dient dann als Programmspeicher. Dabei wird dann das erste REPROM abgeschaltet, damit nicht zwei Speicher gleichzeitig angesprochen werden können.

Ein weiterer Umschalter kann das CMOS-RAM ganz abschalten. Man wird ihn vorzugsweise vor dem Ziehen der Platine betätigen. Er gibt auf den CE1 (Chip-Select-)Eingang H-Pegel und sollte, wie später noch besprochen wird, nur bei gedrückter RESET-Taste umgeschaltet werden.

REPROM-Adreßformat

Bild 26 zeigt das REPROM-Adreßformat. Die Adreß-Leitungen A_9 bis A_0 gehen direkt auf die Adreßeingänge der Bausteine SAB 8708. Diese 10 Bit umfassen ein Adreßvolumen von 1 K = 1024 Adressen. Die Adreß-Leitungen A_{10}, A_{11} und A_{12} führen auf den 1-aus-8-Binär-Decoder SAB 8205. Die beiden Adressen A_{10} und A_{11} werden decodiert. Die Ausgänge 00 bis 03 steuern dann die REPROMs eins bis vier an. Die Adreß-Leitung $\underline{A}_{12}$ führt auf den Freigabeeingang E_1, d. h. der Decoder spricht überhaupt nur dann an, wenn die Adreß-Leitung A_{12} L-Pegel führt. Die Adreß-Leitungen A_{13} bis A_{15} werden nicht benutzt. Sie stehen für andere Zwecke zur Verfügung.

RAM-Adreßformat

Bild 27 zeigt das RAM-Adreßformat. Die Adreß-Leitungen A_7 bis A_0 führen auf die Adreßeingänge der Schreib/Lesespeicher SAB 8111 und auf die gleichen Eingänge der CMOS-RAMs 5101. Die Adreß-Leitungen A_8, A_9, A_{10} führen wieder auf einen 1-aus-8-Decoder SAB 8205. Die Adreß-Leitungen A_8 bis A_{10} werden decodiert und wählen jeweils eines der Schreib/Lesespeicherpaare SAB 8111 oder das CMOS-RAM an. Die Adreß-Leitung A_{12} führt auf den Freigabeeingang E_3; nur, wenn die Adreß-Leitung A_{12} H-Pegel führt, wird also einer der Schreib/Lesespeicher angesprochen. Eine Ausnahme bildet das CMOS-RAM, welches in der Schaltstellung „CMOS-RAM als Programmspeicher" vom Ausgang 0_0 des REPROM-1-aus-8-Decoders angesteuert wird.

Erste Ausbaustufe nur mit CMOS-RAMs

Bild 28 zeigt einen Vorschlag einer ersten Ausbaustufe, die für unsere ersten Versuche vollkommen ausreicht. Die Schaltung Bild 23 bildet einen Aus-

◀ Bild 25. Adressen-Aufteilung des voll bestückten Speichermoduls

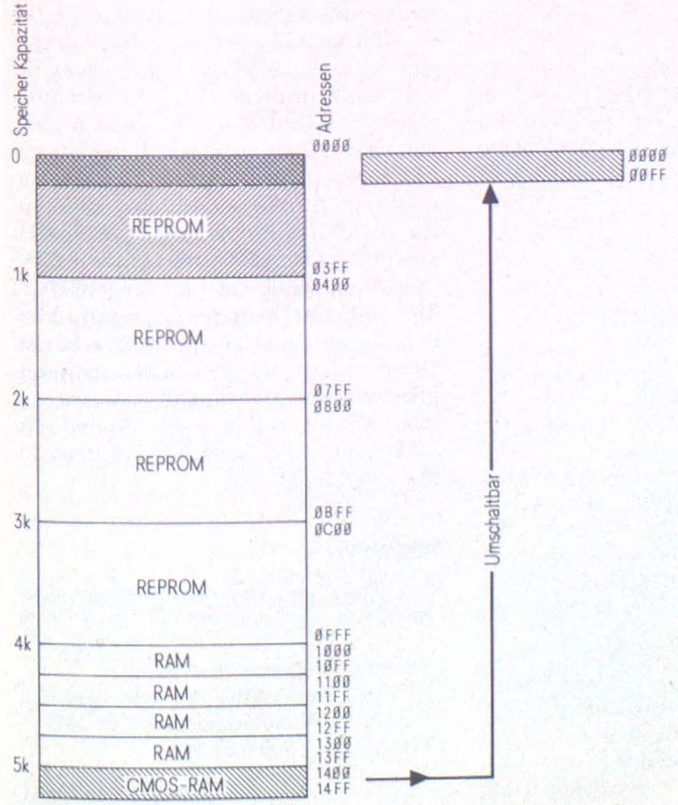
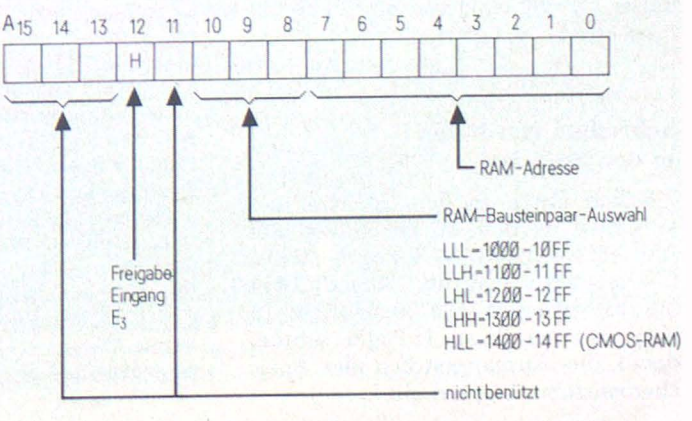


Bild 26. ▼ REPROM-Adreß-Format



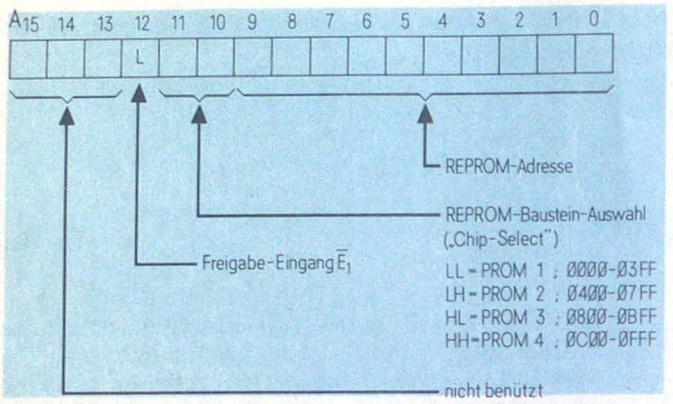


Bild 27. RAM-Adreß-Format

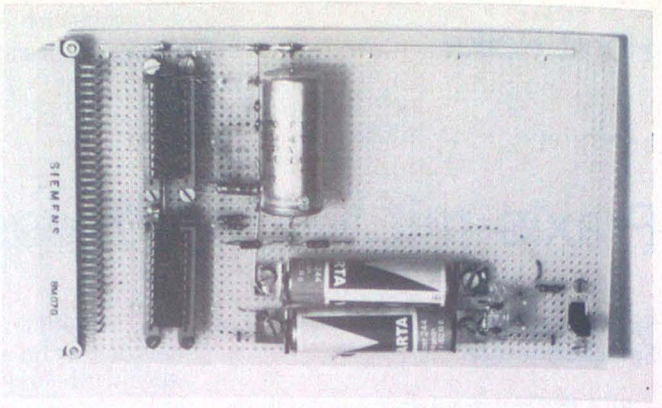


Bild 28. Foto einer ersten Ausbaustufe mit CMOS-RAMs

schnitt aus dem großen Schaltbild Bild 22.

Die Adreßleitungen A_0 bis A_7 führen auf die Adreßeingänge der beiden Bausteine 5101. Die Datenleitungen D_0 bis D_3 führen auf die Dateneinund Ausgänge des oberen Bausteines, die Datenleitungen D_4 bis D_7 auf die Dateneinund Ausgänge des unteren Bausteins.

Damit beim Abschalten der Netzspannung der Speicherinhalt der Speicherzellen nicht verlorengeht, ist die Versorgungsspannung der beiden Bausteine batteriegepuffert. Im Normalbetrieb führt die Speisespannung +5 V über die Siliziumdiode BA 127 auf die Anschlüsse 22 (V_{CC}) der beiden Bausteine. Die Leitung ist mit einem Elektrolyt-Kondensator von 3900 µF abgeblockt. Dieser Kondensator liefert für einige Minuten die Speisespannung weiter, selbst wenn das Netzteil keine +5 V mehr abgibt. Die Langzeitstromversorgung wird von zwei Mignon-Zellen zu 1,5 V übernommen, welche über die Germaniumdiode AA 118 mit dem Elektrolytkondensator verbunden ist.

Lesen des Speicherinhaltes

Um Daten aus dem Speicher lesen zu können, muß die Ausgangssperre „OD" abgeschaltet, d. h. auf L-Pegel liegen. dieser L-Pegel wird von der Steuerleitung $\overline{\text{MEM R}}$ geliefert.

Schreiben von Daten in den Speicher

Sollen Daten in den Speicher geschrieben werden, so bekommen die Anschlüsse 20 (R/W) L-Pegel. Dieser L-Pegel wird von der Steuerleitung $\overline{\text{MEM W}}$ geliefert. Die Anschlüsse 18 (OD) müssen dabei H-Pegel führen, damit die Ausgangsstufen der Speicherbausteine gesperrt sind.

Beim Abschalten der Netzspannung können die vom Mikroprozessor und anderen Stufen kommenden Steuerleitungen $\overline{\text{MEM R}}$ und $\overline{\text{MEM W}}$ zufällig L-Pegel führen, so daß der Speicher dann irgendwelche, während des Abschaltvorganges auf den Datenleitungen befindlichen Zustände in den Speicher einschreibt und nicht den Inhalt behält, der ihm vorher eingeschrieben wurde. Deshalb ist hier eine Sicherheitsschaltung vorgesehen, die im Netzteil mit untergebracht ist. Über die Leitung ZF 0 gelangt normalerweise ein H-Signal an die Freigabeeingänge CE_2. Der CMOS-Speicher wir bereits bei geringfügigem Unterschreiten der Speisespannung 5 V abgeschaltet, die Leitung ZF 0 führt dann L-Pegel.

Adreßleitung A_{12}

Die Freigabeeingänge $\overline{CE_1}$ sind mit den Adreßleitungen A_{12} über den Umschalter verbunden. Die CMOS-RAMs werden nur dann angesprochen, wenn die Adreßleitung A_{12} L-Pegel führt und über den Schalter durchgeschaltet ist. Dies ist bei unserem System dann der Fall, wenn Programmspeicher angesprochen werden sollen. Die CMOS-RAMs sind also hier in diesem Falle als Programmspeicher in das System eingefügt.

Tabelle der Bauteile

2 CMOS-RAM 4 x 256 Bit 5101 L-1 (5101-8) (Fa. Intel)
1 Diode BA 127 (Siemens AG)
1 Germanium-Diode AA 118 (Siemens AG)
1 Germanium-Diode AA 118 (nur wenn CMOS-RAM vom Typ 5101-8
3 Keramik-Kondensatoren 0,1 µF 63 V B 37 449-A6104-S 2 (Siemens AG)
1 Tantal-Perl-Kondensator 2,2 µF, 15 V
1 Elektrolyt-Kondensator 3900 µF, 16 V B 41010-S4398-T1 (Siemens AG)
3 Widerstände 4,7 kΩ
1 Widerstand 47 kΩ
1 einpol. Umschalter (Siemens AG)
1 Steckerleiste DIN 41612 mit 64 Anschlüssen, Bauform C
1 Satz Teile für Fädeltechnik (Siemens AG)

Erste Ausbaustufe mit REPROM SAB 8708

Denjenigen, welche Gelegenheit haben, ein REPROM auf einem Entwicklungsplatz mit PROM-Programmiergerät zu programmieren, sei an dieser Stelle alternativ zum CMOS-RAM eine Lösung mit REPROM SAB 8708 angegeben.

Die Adreßleitungen A_0 bis A_9 sind mit den Adreßeingängen A_0 bis A_9 des Bausteines verbunden, ebenso die Datenleitungen D_0 bis D_7 mit den Ausgängen 0_1 bis 0_8 (Bild 24. Der Eingang für die Bausteinfreigabe $\overline{CS}$ führt zur Steuerleitung $\overline{\text{MEM R}}$. Da in diesem Falle zunächst keine weiteren Speicher angeschlossen sind, wird auch keine zusätzliche Adreßdecodierung benötigt. An Anschluß 24 ist die +5-V-Speisespannung, an Anschluß 19 die +12-V-Speisespannung und an Anschluß 21 die −5-V-Speisespannung anzuschließen, ferner sind die Anschlüsse 12 und 18 zu erden. Die drei Speisespannungen werden zweckmäßigerweise auf dem Modul selbst noch einmal mit Elektrolyt- und Keramik-Kondensatoren abgeblockt.

Stückliste

1 löschbarer und elektrisch neu programmierbarer Festwertspeicher SAB 8708 (Siemens AG)
3 Keramik-Kondensatoren 0,1 µF 63 V B37 449-A 6104-S2 (Siemens AG)
3 Tantal-Perl-Kondensatoren 2,2 µF 15V
1 Widerstand 1 kΩ 0,25 W

(Fortsetzung folgt)

Horst Pelka

Diese Folge behandelt das Einschreiben des Speicherinhaltes in das CMOS-RAM sowie den Mikroprozessor-Modul mit dem Kernstück SAB 8080 und dem quarzgesteuerten Taktgenerator

Praxis mit Mikroprozessoren

6. Teil

Speichermodul
Einschreiben des Speicherinhaltes in das CMOS-RAM mit dem Modul für direkten Speicherzugriff

Bild 29 zeigt das Programm, welches wir für unser erstes elektronisches Spiel benötigen. Es ist so abgebildet, wie es der Entwicklungsplatz, an dem es assembliert ist, ausgedruckt hat. *Tabelle 1* zeigt zum besseren Verständnis die ersten zehn Speicheradressen mit Inhalt der Reihe nach aufgelistet. Die Zahlen der linken Spalte geben die Adressen der Speicherplätze an. In der zweiten Spalte steht der Dateninhalt der Speicherplätze. In den rechten drei Spalten steht das Programm in mnemonischer Form geschrieben. Im Gegensatz dazu sind die ersten beiden Spalten das sogenannte Maschinenprogramm; die erste Spalte enthält die Adressen und die zweite Spalte den Maschinencode für den Mikroprozessor.

In dieser Folge wollen wir auf die Bedeutung der einzelnen Codeworte noch nicht eingehen, sondern Ihnen zeigen, wie Sie dieses Programm mit dem Modul für direkten Speicherzugriff in das CMOS-RAM einschreiben können.

Dazu wird der Schalter „direkter Speicherzugriff" eingeschaltet und mit den Adreßschaltern als erstes die Adresse ϕϕϕϕ eingestellt. Die Datenanzeige kann jetzt einen zunächst beliebigen Speicherinhalt zeigen, nämlich den Speicherinhalt, der zufällig beim Einschalten des Speichers entsteht. Als nächstes stellen wir die Schalter für die Daten auch alle auf ϕ und drücken die Taste „Daten einschreiben". Nach Loslassen dieser Taste muß jetzt die Datenanzeige ebenfalls Null anzeigen.

Das gleiche tun wir für die Adressen ϕϕϕ1 und ϕϕϕ2. In einer späteren Folge wird erklärt, warum wir diese drei Adressen mit dem Befehl NOP, der in Maschinensprache den Code ϕϕ besitzt, eingeschrieben haben. Als nächstes wird die Adresse ϕϕϕ3 eingestellt und die Datenschalter auf 3E geschaltet. Den Zusammenhang zwischen dem Hex-Code und dem Dual-Code fanden Sie in Bild 4 der Folge 1. Jetzt drücken wir wieder die Taste „Daten einschreiben", und die Datenanzeige muß 3E anzeigen.

Die Adresse ϕϕϕ4 ist nicht gesondert im Programm ausgedruckt. Da auf dem Platz ϕϕϕ3 Daten mit zweimal 8 bit ausgedruckt waren, kommt automatisch die 81 auf die Adresse ϕϕϕ4. Es gibt im Mikroprozessor-System 8080 1-Byte-, 2-Byte- und 3-Byte-Befehle. Einen 3-Byte-Befehl finden wir bei Adresse ϕϕ19. Dort sind 3 Bytes Daten vorhanden, die dann an den nachfolgenden Adressen ϕϕ1A und ϕϕ1B eingeschrieben werden müssen.

Jetzt fahren wir also erst mal mit der Adresse ϕϕϕ5 fort und schreiben dort den Inhalt D3 ein, auf die Adresse ϕϕϕ6 kommt der Inhalt EF. Damit fahren wir fort, bis wir das ganze Programm bis zur Adresse ϕϕ5F eingeschrieben haben.

Jetzt können wir beliebige Stichproben machen. Jedesmal, wenn wir eine der linken Adressen einstellen, müssen auf der Datenanzeige die Daten der zweiten Spalte erscheinen.

Bei einem weiteren Versuch schalten wir das Netzteil ab. Lassen Sie das Netzteil ruhig einige Sekunden ausgeschaltet und schalten dann wieder ein. Es müssen nach wie vor im CMOS-RAM alle Daten erhalten geblieben sein. Den gleichen Versuch wiederholen Sie nach einigen Stunden, einigen Tagen oder einigen Wochen. Die Mignon-Zellen müßten, wenn sie frisch waren, etwa ein Jahr halten, um die Daten im CMOS-Speicher aufrechtzuerhalten. Ein Auswechseln der Mignon-Zellen sollte am besten bei eingeschaltetem Netzteil erfolgen. Die Kapazität des 3900-µF-Elkos reicht aber auch aus, um ein Auswechseln der Batterien bei abgeschaltetem Gerät zu überbrücken.

Tabelle 1:

ϕϕϕϕ	ϕϕ
ϕϕϕ1	ϕϕ
ϕϕϕ2	ϕϕ
ϕϕϕ3	3E
ϕϕϕ4	81
ϕϕϕ5	D3
ϕϕϕ6	EF
ϕϕϕ7	3E
ϕϕϕ8	ϕϕ
ϕϕϕ9	D3
ϕϕϕA	EC

Anwender-Programm in REPROM

Steht Ihnen ein Entwicklungsplatz zur Verfügung, so geben Sie die rechten drei Spalten in Assemblersprache in den Entwicklungsplatz ein. Es würde den Rahmen dieser Folge sprengen, wenn wir Ihnen jetzt hier einen kompletten Kursus zur Bedienung eines Entwicklungsplatzes bringen würden. Bitte, nehmen Sie dafür die Bedienungsanleitung für den Entwicklungsplatz zur Hand. Nach Eingabe und Assemblierung bekommen Sie das in Bild 29 abgedruckte komplette Programm mit Adressen und Maschinencode ausgedruckt. Jetzt brauchen Sie nur noch das Programmiergerät aufrufen und entsprechend Anweisung Ihr

```
ISIS 8080 MACRO ASSEMBLER, V1.1
0000                    ORG   0000
0000 00                 NOP
0001 00                 NOP
0002 00                 NOP
0003 3E81               MVI   A,81H
0005 D3EF               OUT   0EFH
0007 3E00               MVI   A,00
0009 D3EC               OUT   0ECH
000B 3E00               MVI   A,00
000D D3ED               OUT   0EDH
000F 0600               MVI   B,00
0011 0E00               MVI   C,00
0013 1601               MVI   D,01H
0015 DBEE     ABF1:     IN    0EEH
0017 E602               ANI   02H
0019 C24300             JNZ   SPI2
001C DBEE               IN    0EEH
001E E608               ANI   08H
0020 CA1500             JZ    ABF1
0023 3EE0     SPI1:     MVI   A,0E0H
0025 D3EE               OUT   0EEH
0027 DBEE     STA1:     IN    0EEH
0029 E601               ANI   01H
002B CA2700             JZ    STA1
002E 7A       SCH1:     MOV   A,D
002F 0F                 RRC
0030 57                 MOV   D,A
0031 DBEE     STO1:     IN    0EEH
0033 E602               ANI   02H
0035 C33B00             JMP   FORT
0038 00                 NOP
0039 00                 NOP
003A 00                 NOP
003B CA2E00   FORT:     JZ    SCH1
003E 7A                 MOV   A,D
003F B0                 ORA   B
0040 D3ED               OUT   0EDH
0042 47                 MOV   B,A
0043 3ED0     SPI2:     MVI   A,0D0H
0045 D3EE               OUT   0EEH
0047 DBEE     STA2:     IN    0EEH
0049 E604               ANI   04H
004B CA4700             JZ    STA2
004E 7A       SCH2:     MOV   A,D
004F 0F                 RRC
0050 57                 MOV   D,A
0051 DBEE     STO2:     IN    0EEH
0053 E608               ANI   08H
0055 CA4E00             JZ    SCH2
0058 7A                 MOV   A,D
0059 B1                 ORA   C
005A D3EC               OUT   0ECH
005C 4F                 MOV   C,A
005D C32300             JMP   SPI1
0000                    END
```

Bild 29. ▶ Anwenderprogramm im Maschinen-Code (links) und mnemonischen Code (rechts)

PROM zu programmieren. In späteren Folgen wird auch auf dieses Problem näher eingegangen. Nachdem Sie jetzt also entweder in Ihrem CMOS-RAM oder in Ihrem REPROM das Anwenderprogramm stehen haben, können Sie mit dem Bau des Mikroprozessor-Moduls fortfahren, denn ohne dieses Anwenderprogramm wäre Ihr Mikroprozessor nicht funktionsfähig.

Literatur

–: Einführung Siemens Microcomputer Entwicklungssystem, System 8080. Bestell-Nr. B 1663

–: Bedienungsanleitung Siemens Microcomputer Entwicklungssystem, System 8080. Best.-Nr. B 1527

–: Hardware-Beschreibung SME PROM-Programmiergerät, System 8080. Best.-Nr. B 1646

Mikroprozessor-Modul

Kernstück des Mikroprozessor-Moduls (*Bild 30*) ist der Mikroprozessor SAB 8080. Zu ihm gehört der Taktgeber und Treiber SAB 8224 sowie der System-Steuer- und Bus-Treiber-Baustein SAB 8228. Diese drei Bausteine ergeben den eigentlichen Mikroprozessor*). Zusätzlich sind auf der Platine sechs invertierende Treiberstufen FLH 481 und sechs Treiberstufen FLH 491 untergebracht. Von den letzteren werden in dieser Ausbaustufe zunächst nur zwei Treiberstufen benutzt. Vorgesehen auf der Platine sind ferner zwei Bausteine SAB 8212 zur Fan-out-Erweiterung des Adreßbusses, welche aber in dieser ersten Ausbaustufe noch nicht benötigt werden. In unserem Falle werden die Adreßausgänge des SAB 8080 direkt mit den Modulanschlüssen verbunden.

Der Taktgenerator befindet sich in dem Baustein 8224. An seine Anschlüsse 14 und 15 sind ein Quarz 18,432 MHz und ein Abgleichkondensator 10 pF angeschlossen. Durch Variation dieses Kondensators von 3...10 pF kann bedarfsweise die genaue Frequenz abgeglichen werden. Der Taktgenerator arbeitet mit der neunfachen Prozessorgeschwindigkeit. Die Geschwindigkeit für den Mikroprozessor beträgt also 0,488 µs. Der Takt wird in die beiden Phasen φ1 und φ2 aufgeteilt und den Anschlüssen 22 und 15 des Mikroprozessors zugeführt.

(Fortsetzung folgt)

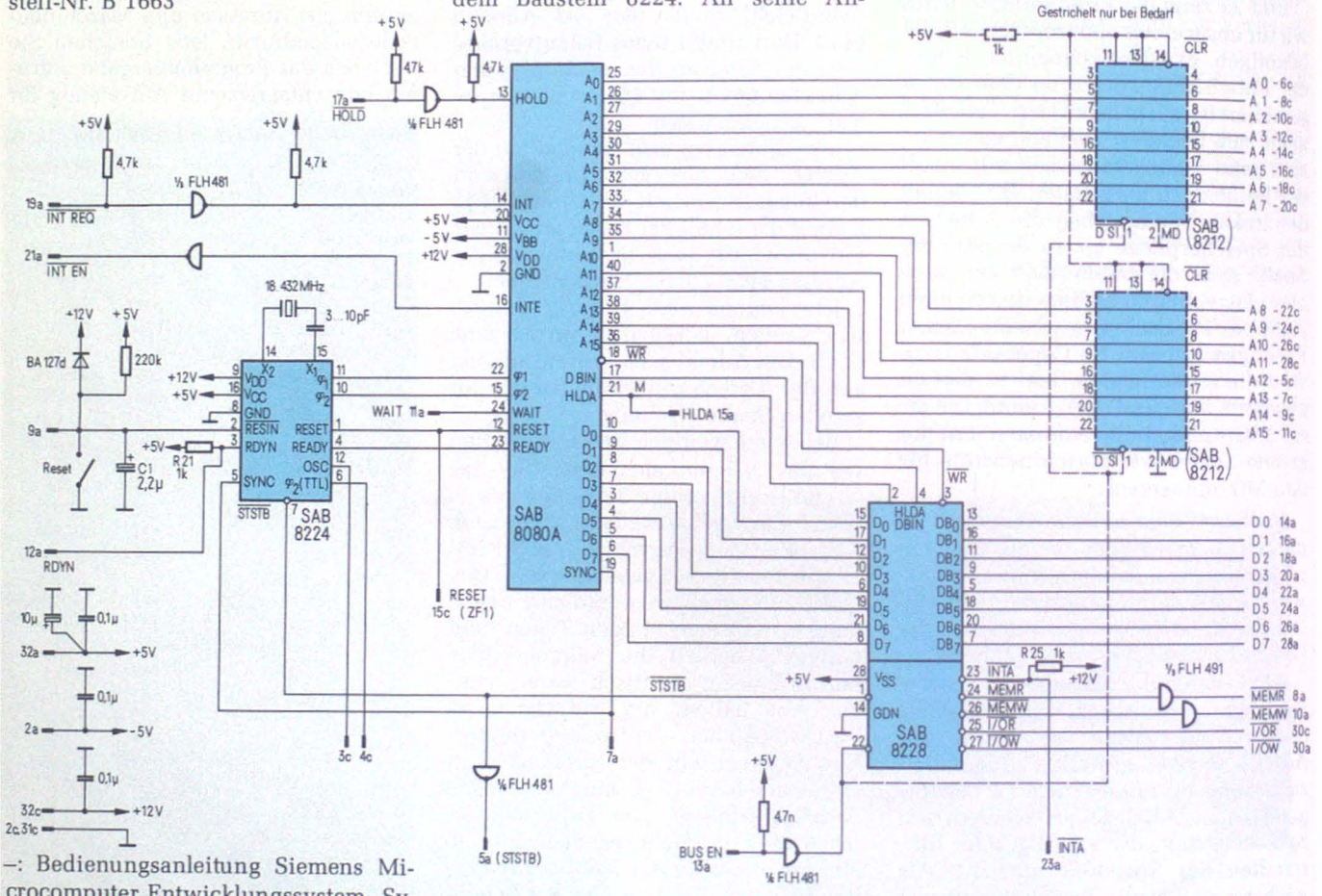
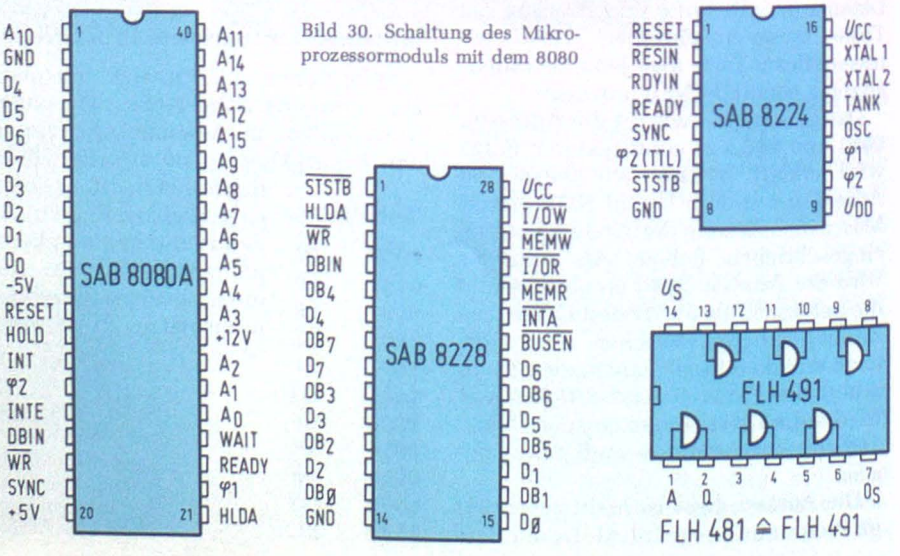


Bild 30. Schaltung des Mikroprozessormoduls mit dem 8080



*) In Kürze wird auch ein anschlußkompatibler Modul mit dem Mikroprozessor 8085 in dieser Serie beschrieben werden, bei dem alle drei Bausteine auf einem Chip integriert sind.

Horst Pelka

Nachdem in der letzten Folge der Aufbau und die Programmierung des Programmspeichers behandelt wurde, kann jetzt mit dem Aufbau des Mikroprozessor-Moduls begonnen werden.

Praxis mit Mikroprozessoren

7. Teil

Der Reset-Eingang (9 a) des Moduls geht auf Anschluß 2 ($\overline{\text{RESIN}}$) des Bausteines 8224. Mit einem RC-Glied 220 kΩ/2,2 µF wird sichergestellt, daß beim Einschalten der Betriebsspannung ein Zwangsreset entsteht. Dies kommt folgendermaßen zustande:

Der Reseteingang ist „aktiv low", d. h. wenn an dem Eingang L-Pegel anliegt, wird auf die Adresse ϕϕϕϕ zurückgestellt. Da vor dem Einschalten die Kapazität C1 keine Ladung besitzt, und diese Ladung erst über den Widerstand 220 kΩ zugeführt wird, wird also im Einschaltmoment L-Pegel an dieser Klemme anliegen. Der Mikroprozessor geht auf die Adresse ϕϕϕϕ.

Gleichzeitig sendet der Mikroprozessor das Befehlsabrufsignal aus *(Bild 31)*. Im Baustein 8228 wird in einer Gatter-Matrix das Steuersignal $\overline{\text{MEM R}}$ (Memory-read – Speicher lesen) erzeugt. Der Speicherplatz mit der Adresse ϕϕϕϕ gibt den Inhalt dieser Speicherzelle auf den Datenbus, dies sind die mit D_0 bis D_7 bezeichneten Leitungen. Im Baustein 8228 befindet sich ein bidirektionaler Datenbuspuffer, die Signale gelangen also über diesen Puffer auf die Anschlüsse D_0 bis D_7 des Mikroprozessors.

In einer Befehlsdecodierschaltung stellt nun der Mikroprozessor fest, welcher Befehl in dem Code enthalten war. Als nächstes erhöht sich der Programmzähler des Mikroprozessors, d. h. es wird die Adresse ϕϕϕ1 ausgegeben. Je nach Art des Befehls in der Adresse ϕϕϕϕ stehen jetzt in der Speicherzelle ϕϕϕ1 entweder Daten oder ein neuer Befehl. Dieser Programmablauf wird in einer späteren Folge noch ausführlicher besprochen werden.

Über den Anschluß 12 a des Moduls kann das Signal „Ready-Input" (RDYN) gegeben werden. Es gelangt über den Anschluß 3 zunächst in den Baustein 8224 und wird als Signal READY an den Mikroprozessor weitergeleitet. Das Bereit-Signal (READY-Signal) zeigt dem 8080 an, daß auf seinem Datenbus Speicher- oder Eingabedaten stehen. Es wird dazu verwendet, den Mikroprozessor mit langsameren Speicher- oder Ein/Ausgabe-Einheiten zu synchronisieren. Wenn der 8080 nach dem Aussenden einer Adresse nicht sofort ein Bereitschaftssignal erhält, geht er in einen Wartezustand über, der solange dauert, wie die Bereitschaftsleitung auf einem L-Pegel verbleibt.

Wir haben bereits auf dem Anzeigemodul eine Hilfsschaltung, die die Ready-Steuerung ausnutzt und mit dem der Mikroprozessor jeweils um einen einzelnen Schritt weitergeschaltet werden kann. Auf diese Art und Weise kann man den Programmablauf langsam verfolgen.

Das Wartesignal „WAIT" bestätigt, daß der Mikroprozessor sich im Wartezustand befindet. Verbindet man nun die Anschlüsse „WAIT" und „READY" miteinander, z. B. über eine Diode AA 118, wie beim CMOS-Speicher beschrieben wurde, so wartet der Mikroprozessor immer eine volle Periode, bis er den nächsten Schritt ausführt. Diese Hilfsschaltung wird immer dann benutzt, wenn der Speicher keine kleine Zugriffszeit hat. Das Warten kommt folgendermaßen zustande: Zunächst wartet ja der Mikroprozessor nicht, und das Signal befindet sich im L-Zustand. L-Zustand zeigt aber dem READY-Eingang an, daß er in den Wartezustand übergehen soll. Ist der Wartezustand erreicht, liegt am WAIT-Ausgang H-Signal an, welches wiederum dem Ready-Eingang des Mikroprozessor anzeigt, daß er jetzt weitermachen kann. Durch den synchronen Ablauf dieser Vorgänge im Mikroprozessor beträgt die Verzögerung dabei eine volle Taktperiode.

An Anschluß 17 a des Moduls kann das Signal $\overline{\text{HOLD}}$ angelegt werden. Es gelangt über einen invertierenden Puffer an den HOLD-Eingang des 8080. Das HOLD-Signal ist eine Anforderung an den Mikroprozessor, in den HOLD-Zustand überzugehen. Dieser Zustand ermöglicht es, die Adreß- und Datenbusse für eine externe Schaltung zu benutzen,

Art des Arbeitsablaufes

Daten BUS-Bit	Zustands-information	Befehlsabruf (1)	Speicher Lesen (2)	Speicher Schreiben (3)	Stapel Lesen (4)	Stapel Schreiben (5)	Eingang Lesen (6)	Ausgang Schreiben (7)	Unterbrechungs-Quittung (8)	Halt-Quittung (9)	Unterbrechungs-Quittung während des Halt-Zustandes (10)
											(N) Zustands-Wort
D_0	INTA	0	0	0	0	0	0	0	1	0	1
D_1	$\overline{\text{WO}}$	1	1	0	1	0	1	0	1	1	1
D_2	STACK	0	0	0	1	1	0	0	0	0	0
D_3	HLTA	0	0	0	0	0	0	0	0	1	1
D_4	OUT	0	0	0	0	0	0	1	0	0	0
D_5	M_1	1	0	0	0	0	0	0	1	0	1
D_6	INP	0	0	0	0	0	1	0	0	0	0
D_7	MEMR	1	1	0	1	0	0	0	0	1	0

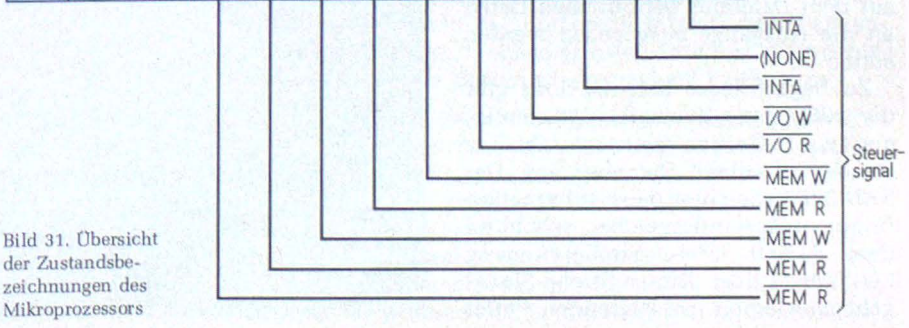


Bild 31. Übersicht der Zustandsbezeichnungen des Mikroprozessors

sobald der 8080 seinen Verkehr mit den Bussen innerhalb des laufenden Befehlszyklus beendet hat. Dies quittiert er an den Ausgang HLDA, welcher an Klemme 15 a des Moduls geführt ist. Dieses Signal zeigt an, daß Daten- und Adreßbusse sich im hochohmigen Zustand befinden. Wir können jetzt z. B. mit dem Modul für direkten Speicherzugriff die einzelnen Speicherzellen auslesen bzw. bei RAM's neue Daten einschreiben.

Über Anschluß 19a des Moduls ($\overline{\text{INT REQ}}$) kann eine Unterbrechungsanforderung an den Mikroprozessor gegeben werden. Dieses Low-Aktiv-Signal führt über einen Inverterpuffer auf den Eingang INT des 8080. Der Mikroprozessor akzeptiert eine Unterbrechungsanforderung auf dieser Leitung entweder am Ende eines laufenden Befehlszyklus oder während eines HOLD-Zustandes. Es muß aber in jedem Falle das Unterbrechungsfreigabe-Flip-flop durch den Befehl EL im Programm gesetzt sein. Wir werden in einer späteren Folge noch Näheres darüber erfahren.

Der Mikroprozessor-Ausgang INTE zeigt den Zustand des internen Unterbrechungs-Freigabe-Flipflop an. Dieses Flipflop kann durch die Befehle EI zur Freigabe oder DI zum Sperren von Unterbrechungen gesetzt, bzw. rückgesetzt werden. Es verhindert im rückgesetzten Zustand die Annahme von Unterbrechungen durch den Mikroprozessor. Das Flipflop wird automatisch während eines Befehlsabrufzyklus zurückgesetzt, wenn ein Unterbrechungsbefehl bereits angenommen wurde. Es wird ebenfalls durch ein Reset-Signal am Rückstelleingang rückgesetzt.

Ist eine Unterbrechungsanforderung vom Mikroprozessor angenommen worden, so gibt der Baustein 8228 das Low-Aktiv-Signal $\overline{\text{INTA}}$ ab.

Weitere Signale sind $\overline{\text{MEM W}}$. Das Signal gibt den Speichern an, daß die Daten des Datenbusses in die Speicherzelle eingeschrieben werden sollen.

Das Signal $\overline{\text{I/O R}}$ gibt den Ein/Ausgabeeinheiten an, daß sie ihre Eingangsdaten an den Datenbus weitergeben sollen, das Signal $\overline{\text{I/O W}}$ bedeutet, daß die auf dem Datenbus befindlichen Daten an die Ausgänge ausgegeben werden sollen.

Zu Beginn jedes Befehlszyklus gibt der 8080 seine Zustands-„Informationen" zur Definition seines internen Zustandes an seinen Datenbus aus. Der SAB 8228 speichert diese Information in einem Zustandsspeicher, sobald an dem $\overline{\text{STSTB}}$ (Status-Strobe)-Eingang LOW-Pegel liegt. Dieses Strobe-Signal geht über einen invertierenden Puffer und wird als STSTB-Signal auch an die Klemme 5 a des Moduls geführt.

Der BUSEN (BUS-ENABLE – Busfreigabe-Eingang) ist ein asynchroner Eingang, der die Datenbusausgänge und Steuersignalausgänge in den potentialfreien Zustand versetzt, wenn L-Signal anliegt. Dieses L-Signal wird in einem invertierenden Puffer invertiert und geht auf Anschluß 22 ($\overline{\text{BUSEN}}$) des 8228.

Zum Synchronisieren anderer Systeme ist noch das Signal SYNC über Klemme 7 a nach außen geführt.

Prüfung des Mikroprozessormoduls

Wir stecken nunmehr den Mikroprozessormodul, den Speichermodul und den Anzeigemodul in die in der Rückwandverdrahtung dafür vorgesehenen Steckerleisten. Der Schalter „Einzelschritte" am Anzeigemodul wird eingeschaltet. Jetzt drücken wir die Reset-Taste am Mikroprozessormodul. Es erscheint die Adresse ϕϕϕϕ. Die Datenanzeige zeigt ϕϕ. Ein Druck auf die Taste „Step" läßt den Mikroprozessor auf die Adresse ϕϕϕ1 gehen. Auch hier sind wieder als Daten ϕϕ eingeschrieben. Ein weiterer Druck auf die Taste „Step" bedeutet den Sprung auf die Adresse ϕϕϕ2 mit ϕϕ im Datenbereich. Die nächste Adresse ϕϕϕ3 hat als Daten 3E eingeschrieben, die Adresse ϕϕϕ4 – 81, die Adresse ϕϕϕ5 – D3, die Adresse ϕϕϕ6 – EF, usw.

Nach der Adresse ϕϕϕ6 erscheint zunächst auf dem Adreßbus EFEF und auf dem Datenbus 81. Hier ist also der Mikroprozessor im nächsten Schritt nicht eine Adresse höher gegangen, sondern hat die im Akkumulator stehende 81 an den Ausgang ausgegeben. Ein- und Ausgänge haben nur eine 1 Byte = 8 Bit lange Adresse; der Mikroprozessor ist so orientiert, daß er diese 8-Bit-Adresse sowohl auf den Adreßbusleitungen A_0 bis A_7 als auch auf den Adreßbusleitungen A_8 bis A_{15} ausgibt.

Beim Durchgehen des Programms in Einzelschritten werden wir also noch häufig Ein- oder Ausgaben zwischen den eigentlichen Adressen bekommen.

Verfolgen wir das Programm nun weiter in Einzelschritten, so wird es vielleicht noch bis zur Adresse ϕϕ22 weitergehen, dann kann es aber eventuell wieder auf die Adresse ϕϕ15 springen. Das hängt davon ab, welche Daten zufällig auf dem Datenbus vorhanden sind. Solange wir den Ein/Ausgabe-Modul und den Bedienteil noch nicht angeschlossen haben, ist der Inhalt des Datenbusses beim Einlesevorgang unbestimmt; sollte er ϕϕ sein, springt das Programm wieder auf Adresse ϕϕ15 zurück, sollte er aber zufällig den Inhalt ϕ8 haben, so können Sie zunächst das Programm noch schrittweise bis Adresse ϕϕ2D weiter verfolgen. Wollen Sie an dieser Stelle sicherstellen, daß kein Rücksprung erfolgt, so müssen Sie die acht Leitungen des Datenbusses mit je 47-kΩ-Widerständen mit +5 V verbinden. Dann geht das Programm zunächst bis Adresse ϕϕ1B, springt von dort nach Adresse ϕϕ43 und läuft bis Adresse ϕϕ5F. Von dort erfolgt ein Sprung nach Adresse ϕϕ23. Die Adressen ϕϕ38 bis ϕϕ3A werden ausgelassen. Das Programm läuft wieder bis Adresse ϕϕ5F und springt zurück nach Adresse ϕϕ23. Die Adressen ϕϕ1C bis ϕϕ22 werden ausgelassen. Sie können nur erreicht werden, wenn bei der auf Adresse ϕϕ16 folgenden Eingabe-Operation das Datenbus-Bit D_1 L-Pegel führt.

Stückliste:

1	Mikroprozessor SAB 8080 A	(Siemens AG)
1	Taktgeber und Treiber SAB 8224	(Siemens AG)
1	System-Steuer- und Bus-Treiber-Baustein SAB 8228	(Siemens AG)
6	invertierende Treiberstufen mit offenem Kollektorausgang FLH 481	(Siemens AG)
6	Treiberstufen mit offenem Kollektorausgang FLH 491	(Siemens AG)
1	Steuerquarz 18,432 MHz Q 87035-A1843-J200	(Siemens AG)
1	Diode BA 127 d	(Siemens AG)
1	Styroflex-Kondensator 10 pF (B 31310-A5100-J)	(Siemens AG)
4	Keramik-Kondensatoren 0,1 µF 63 V B 37449-A6104-SO2	(Siemens AG)
1	Tantal-Perlkondensator 2,2 µF 30 V	
1	Tantal-Perlkondensator 10 µF 15 V	
2	Widerstände 1 kΩ 0,25 W	
5	Widerstände 4,7 kΩ 0,25 W	
1	Widerstand 220 kΩ 0,25 W	
1	einpolige Taste für Reset	
Bei Bedarf:		
2	8-Bit-Eingabe-Ausgabe-Bausteine SAB 8212	(Siemens AG)
1	Widerstand 1 kΩ 0,25 W	

Literatur

Siemens Mikroprozessor Bausteine, System SAB 8080, Datenbuch 1976/77, Best.-Nr. B1526

(Fortsetzung folgt)

In dieser Folge der Beitragsreihe wird zunächst die dynamische Prüfung des Mikroprozessors im Betrieb behandelt. Daran schließt sich die Beschreibung des Ein-/Ausgabemoduls an, der das Bindeglied zwischen dem Datenbus des Mikrocomputers und dem peripheren Aufbau darstellt.

Horst Pelka

Praxis mit Mikroprozessoren

8. Teil

Dynamische Prüfung des Mikroprozessors im Betrieb

Mit der Zusatzschaltung zur Anzeige der Daten in einer vorgewählten Adresse, die wir im Modul für den direkten Speicherzugriff untergebracht haben, können Sie auch den Mikroprozessor im Betrieb verfolgen. Vorausgesetzt, der Datenbus führt im offenen Zustand durch Anschluß der acht Widerstände 47 kΩ die Daten FF, wird der Mikroprozessor beim Drücken der Resettaste zunächst einmal die Adressen ϕϕϕϕ bis ϕϕ5F durchlaufen. Dann springt er wieder auf die Adresse ϕϕ23 zurück und durchläuft den Bereich ϕϕ23 bis ϕϕ5F als Schleife. Wir können nun durch Einstellen einer dieser in Frage kommenden Adressen mit dem Schalter „Speicher/EA" in Stellung „Speicher" die Daten einer jeden Adresse verfolgen.

Bitte, vergessen Sie nicht, wenn Sie mit der „Wahl"-Adresse aus dem Schleifenbereich herausgehen, wird zunächst die Datenanzeige im letzten Zustand verbleiben. Sie müssen jetzt die Taste „Reset" drücken, damit einmal der Mikroprozessor über die von Ihnen angewählte Adresse läuft und so die Daten „geblitzt" werden können. Ähnlich können wir in Stellung EA des Schalters die Eingabe oder Ausgabedaten oder beide gleichzeitig – je nach Schalterstellung – anzeigen. In einer späteren Folge werden wir mit Ihnen zusammen noch einmal das ganze Programm durchgehen. Dies kann jedoch erst dann erfolgen, wenn der Ein-/Ausgabemodul und das Bedienteil fertiggestellt sind. Dann werden wir Ihnen auch die einzelnen Befehle und Schritte genau erklären.

Ein-/Ausgabemodul

Der Ein-/Ausgabemodul stellt das Bindeglied zwischen dem Datenbus des Mikrocomputers und dem peripheren Aufbau dar. Man nennt ihn darum auch Schnittstelle. Da der hier verwendete Baustein 8255 gleichzeitig 1 Byte = 8 Bit parallel ein- oder ausgeben kann, nennt man diese Schnittstelle auch Parallel-Schnittstelle. Insgesamt kann der Baustein drei Bytes (gleich dreimal 8 bit) in drei Kanälen ausgeben, da für die Ausgänge Pufferspeicher vorgesehen sind. In der Betriebsart Null kann aber gleichzeitig immer nur ein Kanal = 8 Bit eingegeben werden, jedoch kurzzeitig hintereinander auch alle drei Kanäle.

Im Gegensatz dazu gibt es auch noch serielle Schnittstellen, bei denen die Informationen hintereinander bitweise ein- oder ausgegeben werden. Man verwendet diese seriellen Schnittstellen z.B. zum Anschluß von Fernschreibern, Druckern, Datensichtgeräten und Tastaturen.

Der hier beschriebene Ein-/Ausgabemodul kann insgesamt mit drei Bausteinen 8255 bestückt werden. Für die erste Ausbaustufe genügt zunächst ein Baustein 8255 (*Bild 32*). Bei dem 8255 handelt es sich um eine programmierbare Schnittstelle. Die funktionellen Eigenschaften werden durch Software bestimmt, so daß normalerweise keine zusätzlichen Logikbausteine erforderlich sind, um periphere Geräte oder Schaltungen anzuschließen.

Ein 8 Bit breiter Zweiwegpuffer mit drei Ausgangszuständen verbindet den internen Datenbus des 8255 mit dem System-Datenbus des Mikrocomputers. Daten werden bei der Ausführung der Befehle Eingabe (IN) und Ausgabe (OUT) vom Puffer abgegeben oder empfangen. Der 8255 enthält ein Steuerregister. Das Steuerwort wird ebenfalls durch den Datenbus übertragen.

Als Ein- und Ausgänge sind drei getrennte Kanäle zu je 8 bit vorhanden. Jeder Kanal und das Steuerwortregister haben eine eigene Adresse. Sie wird mit den beiden niederwertigen Bits des Adreßbusses angesprochen (*Bild 33*). Die Aktivierung des Bausteines erfolgt über den Anschluß $\overline{CS}$. L-Pegel an diesem Eingang veranlaßt den Informationsaustausch zwischen dem 8255 und dem Mikroprozessor. Für die Bausteinauswahl werden die Adreßleitungen A_4 bis A_6 benutzt. Jeweils L-Pegel auf diesen Adreßleitungen aktiviert den zugehörigen Baustein (Bild 33). Über die Steuerbusleitung 30 c gelangt das Lesesignal $\overline{RD}$ an den 8255. L-Pegel an diesem Eingang läßt den Baustein Daten oder Zustandsinformationen über den System-Datenbus an den Mikroprozessor senden. Die Steuerbusleitung 30 a führt auf den Eingang $\overline{WR}$, L-Pegel an diesem Eingang ermöglicht dem Mikroprozessor, Daten oder Steuerworte in den 8255 einzuschreiben.

Die Rücksetzleitung „Reset" (15 c) setzt mit H-Pegel alle Register einschließlich des Steuerregisters zurück und bringt alle Kanäle (A, B, C) in die Betriebsart „Eingabe". Ein entsprechendes Steuerwort, welches im Anwenderprogramm steht, programmiert nun die Betriebsart des 8255 sowie die Konfiguration, ob ein Kanal als Eingabe- oder Ausgabekanal oder bidirektio-

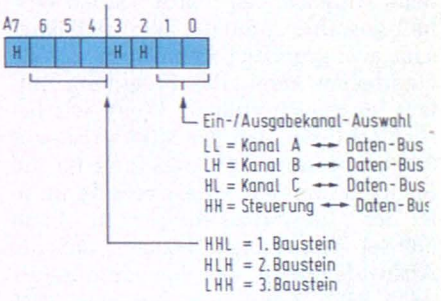


Bild 33. Adreßformat für die Auswahl der Bausteine und Kanäle

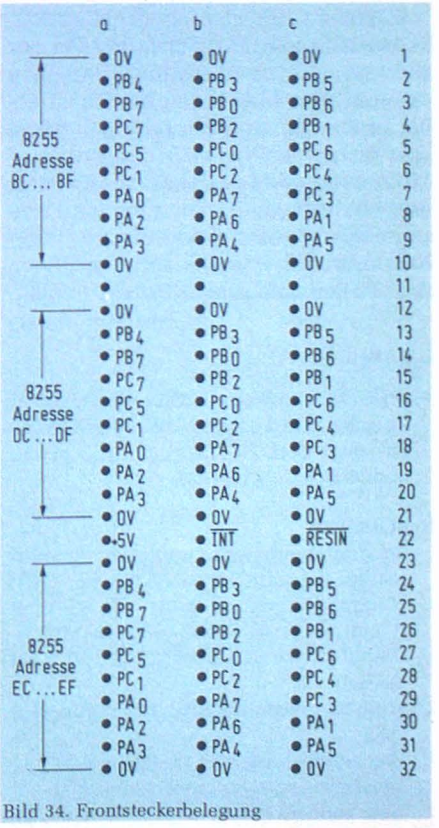


Bild 34. Frontsteckerbelegung

nal benutzt werden soll [7]. In einer späteren Folge werden wir bei der Besprechung des Anwenderprogramms auf diesen Punkt noch ausführlicher eingehen.

Zu beachten ist, daß in Bild 32 auf der linken Seite die Nummern der Steckerleiste für die Rückwandverdrahtung angegeben sind. Die Nummern auf der rechten Seite beziehen sich auf eine Kopfsteckerleiste, über die periphere Geräte an den Mikrocomputer angeschlossen werden können. *Bild 34* zeigt die Steckerbelegung. Hier bleibt es dem Leser überlassen, bedarfsweise einen eigenen Steckertyp, z. B. Tuchel-Stekker, zum Anschluß zu verwenden.

Das in das CMOS-RAM oder REPROM eingeschriebene Programm macht den µC für ein elektronisches Spiel geeignet. Dies ist für einen Mikrocomputer aber nur eine sehr bescheidene Aufgabe. Sie wurde jedoch deshalb gewählt, damit der Leser sich langsam, aber gründlich in die neue Materie einarbeiten kann. Das Programm umfaßt 95 Speicherplätze. Wenn wir berücksichtigen, daß der Mikroprozessor 8080 insgesamt 64 k, das sind 65 536 Speicheradressen, adressieren kann, so ist der Mikroprozessor nur zu einem 689stel ausgelastet. Bezogen auf die Ausbaufähigkeit des Speichermoduls nach Bild 22 (Folge 4), bei dem 5376 Speicherplätze belegt werden können, haben wir jetzt ein 56stel verbraucht.

Es stehen uns also noch viele Möglichkeiten offen. Zunächst müssen wir aber eine Kommunikation zwischen uns und dem Mikrocomputer schaffen. Bei großen Computern oder bei einem sehr fortgeschrittenen Ausbau unseres Mikrocomputers sind dies eine Tastatur und ein Sichtgerät oder auch ein Fernschreiber. Für das elektronische Unterhaltungsspiel „Seeschlacht" genügen fünf Tasten und einige Leuchtdioden.

(Fortsetzung folgt)

Literatur

[7] *Pelka, H.*: Umgang mit Mikroprozessoren; Eingabe- und Ausgabebausteine. Siemens Bauteilereport 15 (1977) 2, S. 38 bis 43 und Heft 3, S. 119...122.

Bauteileliste:

1 programmierbarer peripherer Schnittstellen-Baustein SAB 8255 (Siemens AG)
1 Keramik-Kondensator 0,1 µF, 63 V, B 37449 – A 6104 – SO2 (Siemens AG)
1 Tantal-Perlkondensator 2,2 µF, 30 V bei Bedarf:
2 weitere Bausteine SAB 8255 (Siemens AG)
1 Steckerleiste 64polig (Siemens AG)
1 Steckerleiste 96polig (Siemens AG)
1 Satz Teile für Fädeltechnik (Siemens AG)

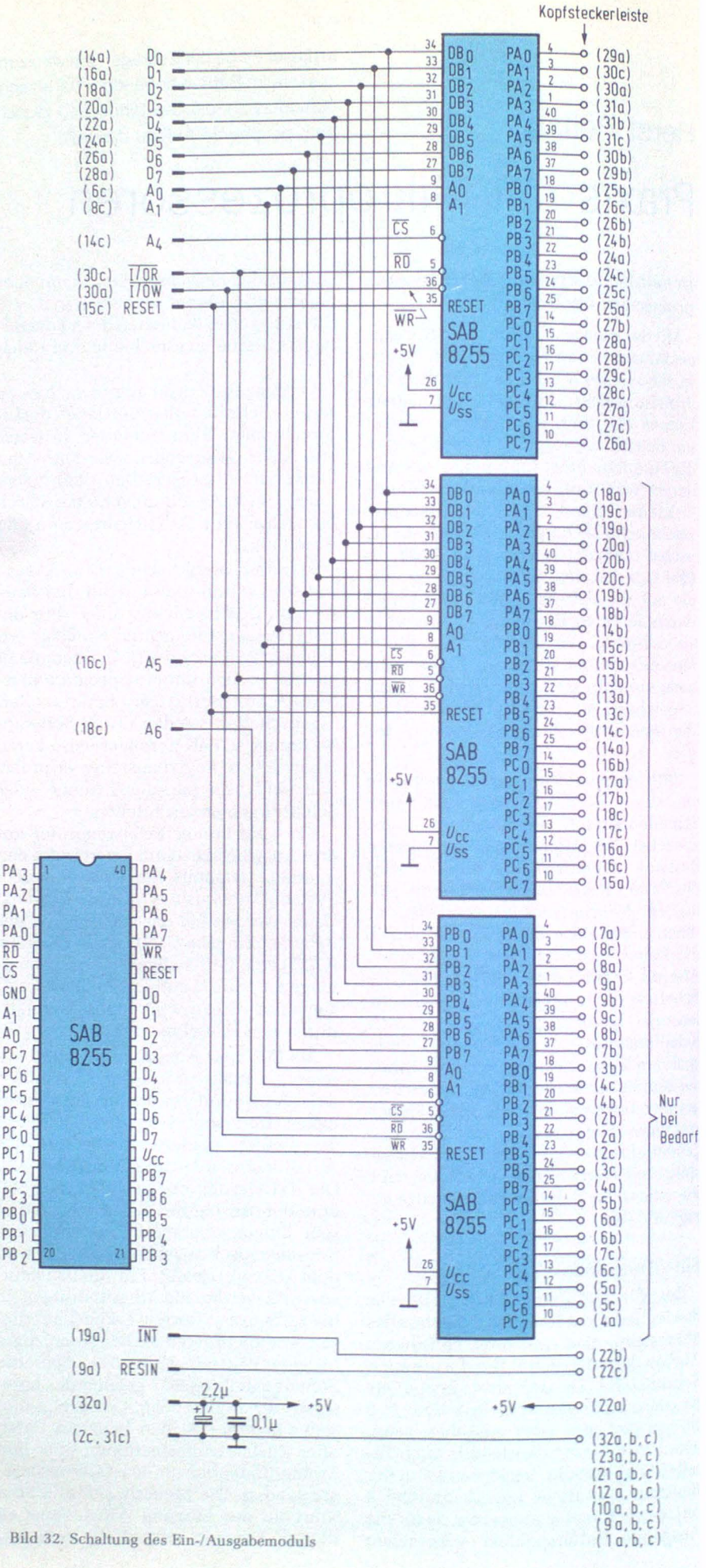


Bild 32. Schaltung des Ein-/Ausgabemoduls

Horst Pelka

Nach Fertigstellung des Ein-/Ausgabemoduls sind wir nach der Begriffsbestimmung (siehe Praxis mit Mikroprozessor, Teil 1) im Besitz eines Mikrocomputers.

Praxis mit Mikroprozessoren

9. Teil

Bedienteil

Das Bedienteil besteht im wesentlichen nur aus Tasten und Anzeigen (*Bild 35*). An den Kanal A (P_{A0}... P_{A7}) sind sechs grüne Leuchtdioden LD 57 CA angeschlossen. Zwei Klemmen bleiben frei. An den Kanal B (P_{B0}... P_{B7}) kommen sechs rote Leuchtdioden. Ferner sind zwei gelbe Leuchtdioden LD 56 CA mit den Klemmen P_{C4} und P_{C5} des Kanals C verbunden. Die gelben Leuchtdioden sollen später den Spielern signalisieren, wer in dem Spiel an der Reihe ist. Alle Leuchtdioden haben Vorwiderstände von 470 Ω, welche den Strom durch die Dioden auf etwa 6 oder 7 mA einstellen. Der Strom fließt von der Speisespannung +5 V über Leuchtdioden und Vorwiderstände in die Klemmen der Kanäle des Bausteins 8255 hinein. Ein Strom in dieser Größenordnung ist durchaus zulässig, wenn man eine höhere Restspannung als üblicherweise 0,4 V bei L-Pegel zuläßt.

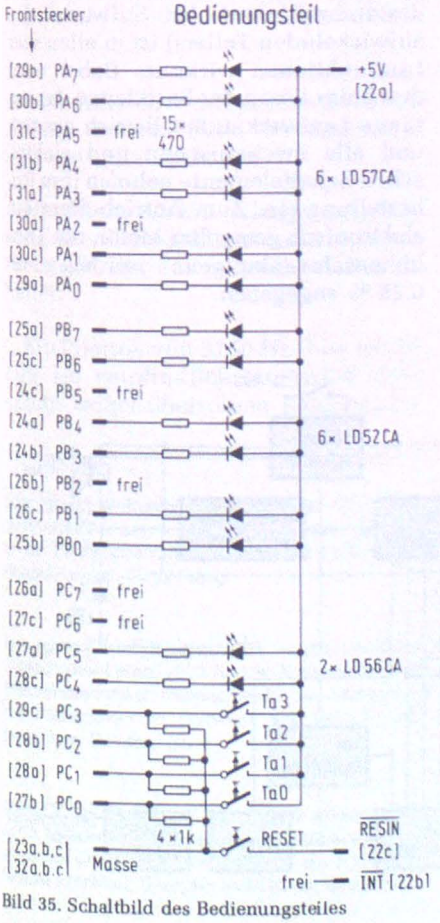


Bild 35. Schaltbild des Bedienungsteiles

Die Klemmen P_{C0}... P_{C3} des Kanales C sind über Widerstände von 1 kΩ mit Masse verbunden und führen auf die Tasten TA 0...TA 3. Bei gedrückter Taste sind die Klemmen mit der Speisespannung +5 V verbunden. Eine weitere Taste „RESET" führt von Masse nach Anschluß 22c der Frontsteckerleiste. Sämtliche in Bild 35 in eckigen Klammern angegebenen Stiftbelegungen beziehen sich auf den Frontstecker des Ein-/Ausgabemoduls, also *nicht* auf die Rückwandverdrahtung aller Moduln.

Bild 36 zeigt einen Vorschlag für die Gestaltung der Deckplatte des Bedienungsteiles. Sie kann ähnlich gestaltet werden wie in der Veröffentlichung der ELO, nur daß jetzt noch die beiden gelben Leuchtdioden hinzukommen.

Nach Drückung der Taste „RESET" leuchten alle Leuchtdioden, auch die beiden gelben, auf. Durch Drücken der Taste „Treffer" kann einer der beiden Spieler seinem Gegner gestatten, anzufangen. Dann drückt der Gegner zunächst die Taste „Schuß" und danach die Taste „Treffer". Je nach dem, ob ein Treffer vorliegt, erlischt eine der gegnerischen Leuchtdioden. Gleichzeitig leuchtet beim Gegner die gelbe Leuchtdiode auf, als Zeichen, daß er jetzt an der Reihe ist.

Die grünen Leuchtdioden sind dem Kanal A zugeordnet, die roten dem Kanal B. Welche der sechs Leuchtdioden mit welcher Klemme verbunden wird, ist an sich gleichgültig, nur sollte die Beschaltung für die Kanäle A und B gleich sein, d. h. die Leuchtdiode mit einer bestimmten Wertung sollte im Kanal A und im Kanal B mit der Klemme gleicher Wertigkeit verbunden werden, damit beiden Spielern die gleichen Gewinnchancen zukommen.

Damit ist zunächst der Aufbau der „Hardware" abgeschlossen. In der nächsten Fortsetzung wird das Anwenderprogramm erklärt.

(Fortsetzung folgt)

Bauteileliste:

6 Leuchtdioden LD 52 CA (Siemens AG)
2 Leuchtdioden LD 56 CA (Siemens AG)
6 Leuchtdioden LD 57 CA (Siemens AG)
15 Widerstände 470 Ω
4 Widerstände 1 kΩ
5 einpolige Tasten
1 Steckerleiste für Frontstecker
1 mindestens 21poliges Verbindungskabel (kann auch aus Einzellitzen zusammengesetzt werden)
1 geeignetes Gehäuse

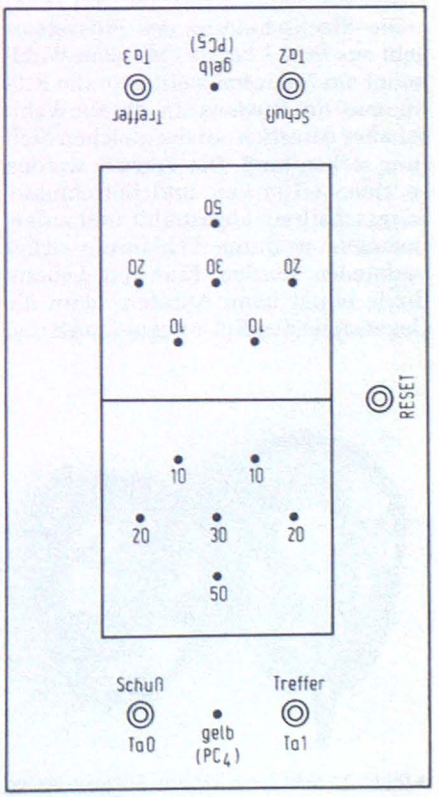


Bild 36. Vorschlag für die Gestaltung der Deckplatte des Bedienungsteiles. Die Ziffern neben den Leuchtdioden geben die Wertigkeit der „Einheiten" an. Sie können durch Skizzen von Schlachtschiffen, Zerstörern, Flugzeugträgern und Schnellbooten noch ergänzt werden

Berichtigung Praxis mit Mikroprozessoren

1. Teil, FUNKSCHAU 1977, Heft 15, Seite 676, rechte Spalte Zeile 41: 1,2 kΩ (nicht 820 Ω).
2. Teil, Seite 724, Bild 13: Beim oberen Baustein 8212 ist es Anschluß 10 (nicht 18), der mit Adreßleitung A_{11}, Anschluß 28c verbunden ist. Die untere Diode AA 118 führt rechts zu Anschluß 17c (nicht 12c).
3. Teil, Heft 17, Seite 802, Bild 20: – 5 V ist 2a (nicht 2c)
4. Teil, Seite 862, Bild 23, links oben: Leitung $\overline{\text{MEM R}}$ ist 8a (nicht 8c) Leitung $\overline{\text{MEM W}}$ ist 10a (nicht 10c)
6. Teil, Heft 20, Seite 946, Bild 30: φ_2 TTL in 4a (nicht 4c).

Horst Pelka

Mit dem Aufbau des Bedienungsteiles ist die Erstellung der „Hardware" abgeschlossen. Nun wenden wir uns der Erstellung bzw. Erklärung der „Software" zu. Die „Software" umfaßt alle Programme, die in einem Computer gebraucht werden. Hier in unserem Falle handelt es sich um ein Anwenderprogramm, ferner gibt es noch Betriebsprogramme, Entwicklungsprogramme und Simulatorprogramme.

Praxis mit Mikroprozessoren

10. Teil

Tabelle 1 zeigt den Befehlssatz des Mikroprozessors 8080, geordnet nach Befehlsarten, *Tabelle 2* den gleichen Befehlssatz, geordnet nach dem Maschinencode.

Die genaue Beschreibung der einzelnen Befehle finden Sie in [3].

Der Maschinencode ist eine zweistellige Sedezimalzahl (Hexcode). Man unterscheidet 1 Byte lange Befehle, 2 Byte-Befehle und 3 Byte-Befehle. 1 Byte lange Befehle sind immer dann gegeben, wenn es sich um einen einfachen Befehl ohne zusätzliche Daten- oder Adressenangaben handelt. Ist zusätzlich zu dem eigentlichen Befehl noch eine Datenangabe erforderlich, so wird ein weiteres Byte = 8 Bit für die Daten gebraucht, und es handelt sich um einen 2 Byte-Befehl. Werden zusätzlich zu dem Befehl Daten mit 2 Byte = 16 Bit gebraucht, oder zusätzlich zu dem Befehl wird eine Adresse gebraucht, so braucht man für die Adresse noch einmal 2 Bytes. Es handelt sich dann um 3-Byte-Befehle. Eine Ausnahme bilden der Eingabe- und Ausgabe-Befehl, da in diesem Falle die Adresse nur 8 Bit = 1 Byte lang ist; Ein- und Ausgabebefehle sind 2-Byte-Befehle.

Liest der Mikroprozessor einen Befehl aus dem Speicher, so gelangt zunächst ein Byte in den Befehlsdecoder. Der Befehlsdecoder decodiert den Befehl, und die logische Steuerung im Mikroprozessors weiß sofort aus der Art des Befehles, ob es sich um einen 1, 2 oder 3 Byte langen Befehl handelt. Bei einem 1 Byte langen Befehl wird also

Tabelle 1

Transfer-Befehle

40	MOV	B,B
41	MOV	B,C
42	MOV	B,D
43	MOV	B,E
44	MOV	B,H
45	MOV	B,L
46	MOV	B,M
47	MOV	B,A
48	MOV	C,B
49	MOV	C,C
4A	MOV	C,D
4B	MOV	C,E
4C	MOV	C,H
4D	MOV	C,L
4E	MOV	C,M
4F	MOV	C,A
50	MOV	D,B
51	MOV	D,C
52	MOV	D,D
53	MOV	D,E
54	MOV	D,H
55	MOV	D,L
56	MOV	D,M
57	MOV	D,A
58	MOV	E,B
59	MOV	E,C
5A	MOV	E,D
5B	MOV	E,E
5C	MOV	E,H
5D	MOV	E,L
5E	MOV	E,M
5F	MOV	E,A
60	MOV	H,B
61	MOV	H,C
62	MOV	H,D
63	MOV	H,E
64	MOV	H,H
65	MOV	H,L
66	MOV	H,M
67	MOV	H,A
68	MOV	L,B
69	MOV	L,C
6A	MOV	L,D
6B	MOV	L,E
6C	MOV	L,H
6D	MOV	L,L
6E	MOV	L,M
6F	MOV	L,A
70	MOV	M,B
71	MOV	M,C
72	MOV	M,D
73	MOV	M,E
74	MOV	M,H
75	MOV	M,L
–	–	–
77	MOV	M,A
78	MOV	A,B
79	MOV	A,C
7A	MOV	A,D
7B	MOV	A,E
7C	MOV	A,H
7D	MOV	A,L
7E	MOV	A,M
7F	MOV	A,A

Doppel-Register

0A	LDAX	B
1A	LDAX	D
2A	LHLD	Adr
3A	LDA	Adr
02	STAX	B
12	STAX	D
22	SHLD	Adr
32	STA	Adr

–,direkte 8Bit

06	MVI	B,	D8
0E	MVI	C,	
16	MVI	D,	
1E	MVI	E,	
26	MVI	H,	
2E	MVI	L,	
36	MVI	M,	
3E	MVI	A,	

–,direkte 16Bit

01	LXI	B,	D16
11	LXI	D,	
21	LXI	H,	
31	LXI	SP,	

Arithmetische Operationen

80	ADD	B
81	ADD	C
82	ADD	D
83	ADD	E
84	ADD	H
85	ADD	L
86	ADD	M
87	ADD	A
88	ADC	B
89	ADC	C
8A	ADC	D
8B	ADC	E
8C	ADC	H
8D	ADC	L
8E	ADC	M
8F	ADC	A
90	SUB	B
91	SUB	C
92	SUB	D
93	SUB	E
94	SUB	H
95	SUB	L
96	SUB	M
97	SUB	A
98	SBB	B
99	SBB	C
9A	SBB	D
9B	SBB	E
9C	SBB	H
9D	SBB	L
9E	SBB	M
9F	SBB	A

Akkumulator rotieren

07	RLC	
0F	RRC	
17	RAL	
1F	RAR	

Doppel-Register

09	DAD	B
19	DAD	D
29	DAD	H
39	DAD	SP

–, direkte 8 Bit

C6	ADI	D8
CE	ACI	
D6	SUI	
DE	SBI	
E6	ANI	
EE	XRI	
F6	ORI	
FE	CPI	

Inkrementieren

04	INR	B
0C	INR	C
14	INR	D
1C	INR	E
24	INR	H
2C	INR	L
34	INR	M
3C	INR	A
03	INX	B
13	INX	D
23	INX	H
33	INX	SP

Dekrementieren

05	DCR	B
0D	DCR	C
15	DCR	D
1D	DCR	E
25	DCR	H
2D	DCR	L
35	DCR	M
3D	DCR	A
0B	DCX	B
1B	DCX	D
2B	DCX	H
3B	DCX	SP

Logische Operationen

A0	ANA	B
A1	ANA	C
A2	ANA	D
A3	ANA	E
A4	ANA	H
A5	ANA	L
A6	ANA	M
A7	ANA	A
A8	XRA	B
A9	XRA	C
AA	XRA	D
AB	XRA	E
AC	XRA	H
AD	XRA	L
AE	XRA	M
AF	XRA	A

B0	ORA	B
B1	ORA	C
B2	ORA	D
B3	ORA	E
B4	ORA	H
B5	ORA	L
B6	ORA	M
B7	ORA	A
B8	CMP	B
B9	CMP	C
BA	CMP	D
BB	CMP	E
BC	CMP	H
BD	CMP	L
BE	CMP	M
BF	CMP	A

Eingabe/Ausgabe

D3	OUT	D8
DB	IN	D8

Sprungbefehle

C3	JMP	Adr
C2	JNZ	Adr
CA	JZ	Adr
D2	JNC	Adr
DA	JC	Adr
E2	JPO	Adr
EA	JPE	Adr
F2	JP	Adr
FA	JM	Adr
E9	PCHL	

Programmaufrufe

CD	CALL	Adr
C4	CNZ	Adr
CC	CZ	Adr
D4	CNC	Adr
DC	CC	Adr
E4	CPO	Adr
EC	CPE	Adr
F4	CP	Adr
FC	CM	Adr

Rücksprungbefehle

C9	RET	
C0	RNZ	
C8	RZ	
D0	RNC	
D8	RC	
E0	RPO	
E8	RPE	
F0	RP	
F8	RM	

Kellerspeicher-Operationen

C5	PUSH	B
D5	PUSH	D
E5	PUSH	H
F5	PUSH	PSW
C1	POP	B
D1	POP	D
E1	POP	H
F1	POP	PSW*
E3	XTHL	
F9	SPHL	

Steuerbefehle

00	NOP	
76	HLT	
F3	DI	
FB	EI	

Restart

C7	RST	0
CF	RST	1
D7	RST	2
DF	RST	3
E7	RST	4
EF	RST	5
F7	RST	6
FF	RST	7

Sonstige

EB	XCHG	
27	DAA*	
2F	CMA	
37	STC+	
3F	CMC+	

D 8 = 8 Bit Daten
D16 = 16 Bit Daten
Adr = 16 Bit Adresse
+ Beeinflußt nur Carry-Bit
* Beeinflußt alle Flags

Tabelle 2

00	NOP		2B	DCX	H	56	MOV	D,M	81	ADD	C	AC	XRA	H	D5	PUSH	D
01	LXI	B,D16	2C	INR	L	57	MOV	D,A	82	ADD	D	AD	XRA	L	D6	SUI	D8
02	STAX	B	2D	DCR	L	58	MOV	E,B	83	ADD	E	AE	XRA	M	D7	RST	2
03	INX	B	2E	MVI	L,D8	59	MOV	E,C	84	ADD	H	AF	XRA	A	D8	RC	
04	INR	B	2F	CMA		5A	MOV	E,D	85	ADD	L				D9	–	
05	DCR	B	30	–		5B	MOV	E,E	86	ADD	M	B0	ORA	B	DA	JC	Adr
06	MVI	B,D8	31	LXI	SP,D16	5C	MOV	E,H	87	ADD	A	B1	ORA	C	DB	IN	D8
07	RLC		32	STA	Adr	5D	MOV	E,L	88	ADC	B	B2	ORA	D	DC	CC	Adr
08	–		33	INX	SP	5E	MOV	E,M	89	ADC	C	B3	ORA	E	DD	–	
09	DAD	B	34	INR	M	5F	MOV	E,A	8A	ADC	D	B4	ORA	H	DE	SBI	D8
0A	LDAX	B	35	DCR	M	60	MOV	H,B	8B	ADC	E	B5	ORA	L	DF	RST	3
0B	DCX	B	36	MVI	M,D8	61	MOV	H,C	8C	ADC	H	B6	ORA	M	E0	RPO	
0C	INR	C	37	STC		62	MOV	H,D	8D	ADC	L	B7	ORA	A	E1	POP	H
0D	DCR	C	38	–		63	MOV	H,E	8E	ADC	M	B8	CMP	B	E2	JPO	Adr
0E	MVI	C,D8	39	DAD	SP	64	MOV	H,H	8F	ADC	A	B9	CMP	C	E3	XTHL	
0F	RRC		3A	LDA	Adr	65	MOV	H,L	90	SUB	B	BA	CMP	D	E4	CPO	Adr
10	–		3B	DCX	SP	66	MOV	H,M	91	SUB	C	BB	CMP	E	E5	PUSH	H
11	LXI	D,D16	3C	INR	A	67	MOV	H,A	92	SUB	D	BC	CMP	H	E6	ANI	D8
12	STAX	D	3D	DCR	A	68	MOV	L,B	93	SUB	E	BD	CMP	L	E7	RST	4
13	INX	D	3E	MVI	A,D8	69	MOV	L,C	94	SUB	H	BE	CMP	M	E8	RPE	
14	INR	D	3F	CMC		6A	MOV	L,D	95	SUB	L	BF	CMP	A	E9	PCHL	
15	DCR	D	40	MOV	B,B	6B	MOV	L,E	96	SUB	M				EA	JPE	Adr
16	MVI	D,D8	41	MOV	B,C	6C	MOV	L,H	97	SUB	A	C0	RNZ		EB	XCHG	
17	RAL		42	MOV	B,D	6D	MOV	L,L	98	SBB	B	C1	POP	B	EC	CPE	Adr
18	–		43	MOV	B,E	6E	MOV	L,M	99	SBB	C	C2	JNZ	Adr	ED	–	
19	DAD	D	44	MOV	B,H	6F	MOV	L,A	9A	SBB	D	C3	JMP	Adr	EE	XRI	D8
1A	LDAX	D	45	MOV	B,L	70	MOV	M,B	9B	SBB	E	C4	CNZ	Adr	EF	RST	5
1B	DCX	D	46	MOV	B,M	71	MOV	M,C	9C	SBB	H	C5	PUSH	B	F0	RP	
1C	INR	E	47	MOV	B,A	72	MOV	M,D	9D	SBB	L	C6	ADI	D8	F1	POP	PSW
1D	DCR	E	48	MOV	C,B	73	MOV	M,E	9E	SBB	M	C7	RST	0	F2	JP	Adr
1E	MVI	E,D8	49	MOV	C,C	74	MOV	M,H	9F	SBB	A	C8	RZ		F3	DI	
1F	RAR		4A	MOV	C,D	75	MOV	M,L	A0	ANA	B	C9	RET	Adr	F4	CP	Adr
20	–		4B	MOV	C,E	76	HLT		A1	ANA	C	CA	JZ		F5	PUSH	PSW
21	LXI	H,D16	4C	MOV	C,H	77	MOV	M,A	A2	ANA	D	CB	–		F6	ORI	D8
22	SHLD	Adr	4D	MOV	C,L	78	MOV	A,B	A3	ANA	E	CC	CZ	Adr	F7	RST	6
23	INX	H	4E	MOV	C,M	79	MOV	A,C	A4	ANA	H	CD	CALL	Adr	F8	RM	
24	INR	H	4F	MOV	C,A	7A	MOV	A,D	A5	ANA	L	CE	ACI	D8	F9	SPHL	
25	DCR	H	50	MOV	D,B	7B	MOV	A,E	A6	ANA	M	CE	RST	1	FA	JM	Adr
26	MVI	H,D8	51	MOV	D,C	7C	MOV	A,H	A7	ANA	A				FB	EI	
27	DAA		52	MOV	D,D	7D	MOV	A,L	A8	XRA	B	D0	RNC		FC	CM	Adr
28	–		53	MOV	D,E	7E	MOV	A,M	A9	XRA	C	D1	POP	D	FD	–	
29	DAD	H	54	MOV	D,H	7F	MOV	A,A	AA	XRA	D	D2	JNC	Adr	FE	CPI	D8
2A	LHLD	Adr	55	MOV	D,L	80	ADD	B	AB	XRA	E	D3	OUT	D8	FF	RST	7
												D4	CNC	Adr			

der Befehl vom Mikroprozessor ausgeführt. Handelt es sich um einen 2 Byte langen Befehl, so gibt der Mikroprozessor einen weiteren Lesebefehl für das nachfolgende Byte automatisch aus. Ist der Befehl 3 Bytes lang, so werden also nach dem eigentlichen Befehl. der decodiert wird, gleichzeitig noch zwei weitere Bytes gelesen. Aus dem Maschinencode weiß der Mikroprozessor, ob es sich um Daten oder Adressen handelt.

Erstellung eines Anwenderprogramms

Ehe man an die Erstellung eines Programms herangeht, wird ein Flußdiagramm bzw. ein Ablaufplan erstellt. Flußdiagramme ermöglichen die anschauliche und übersichtliche Darstellung komplexer Zusammenhänge. Die Sinnbilder für das Flußdiagramm sind in der Norm DIN 66001 festgelegt. Näheres darüber finden Sie in [3].

Bild 37 zeigt das Flußdiagramm für unser Anwenderprogramm. Wir wollen jetzt anhand dieses Flußdiagramms und der Einzelschritte das Programm (siehe Bild 29, Folge 6) verfolgen.

Anwenderprogramm

Stellen Sie bitte den Schalter „Einzelschritte" am Anzeigemodul auf „Single-step". Jetzt drücken Sie die Reset-Taste, es wird oben die

Adresse ØØØØ mit den Daten ØØ

angezeigt. Es handelt sich um den Befehl NOP (no operation), welcher den Maschinencode ØØ hat. Diesen Befehl benutzt man, wenn man eine Speicherstelle für andere Zwecke freihalten oder eine Zeitverzögerung programmieren möchte. Dieser Befehl „NOP" ist hier am Anfang dreimal vorhanden, d. h. drei Bytes sind freigehalten worden, um später dort einen 3-Byte-Befehl unterbringen zu können. Dieser 3-Byte-Befehl wäre z. B. ein Sprungbefehl auf eine andere Adresse, so daß der Mikroprozessor nicht das nachfolgende Programm abarbeitet, sondern ein auf einer anderen Adresse stehendes neues Programm.

Nach Drücken der Taste „Reset" verlöschen die Leuchtdioden im Anzeigeteil, weil das Reset-Signal gleichzeitig das Steuerwortregister im Ein/Ausgabe-Baustein 8255 rücksetzt und sämtliche Kanäle als Eingabe-Kanäle konfiguriert. Damit können die Lampen nicht mehr angesteuert werden. Eine genaue Beschreibung der Konfiguration der Ein/Ausgabe-Kanäle finden Sie in [3].

Wir fahren jetzt mit der Besprechung des Programms fort, drücken die Taste „Step", es erscheint die

Adresse ØØØ1 mit den Daten ØØ.

Ein weiterer Druck auf die Taste bringt uns zur

Adresse ØØØ2 mit den Daten ØØ.

Wie bereits besprochen, handelt es sich um drei Befehle „No operation"; die Speicherstellen wurden für später freigehalten. Ein weiterer Druck auf die Taste „Step" bringt uns zur

Adresse ØØØ3 mit den Daten 3E.

3E bedeutet „MVI A, D8", d. h. in den Akkumulator sollen 8 Bit-Daten eingeschrieben werden. In der nachfolgenden Adresse werden also die Daten stehen, die in den Akkumulator kommen sollen. Wir drücken die Taste „Step" und kommen zur

Adresse ØØØ4 mit den Daten 81.

Bei der 81 handelt es sich um ein Bit-Muster zur Festlegung der Betriebsart des Ein/Ausgabe-Bausteines 8255 im Ein/Ausgabe-Modul und um die Festlegung, welche Kanäle als Eingabe und welche Kanäle als Ausgabe fungieren sollen. In diesem Fall wird der Baustein für die Betriebsart Ø programmiert, wobei die Kanäle A, B und die vier höherwertigen Bits des Kanals C als Ausgaben arbeiten, als Eingaben die vier niederwertigen Bits des Kanales C. Wir müssen nun dieses Steuerwort an den Ein/Ausgabe-Baustein 8255 senden; dieses machen wir mit einem Ausgabebefehl. Auf der

Adresse ØØØ5 steht D3.

D3 bedeutet „OUT", also einen Ausgabebefehl; im nachfolgenden Byte muß angegeben werden, wohin die Daten kommen sollen. Auf der

Adresse ØØØ6 steht EF.

Die Adresse „EF" ist dem Steuerwortregister des Ein/Ausgabe-Bausteines 8255 zugeteilt. Ein erneuter Druck auf

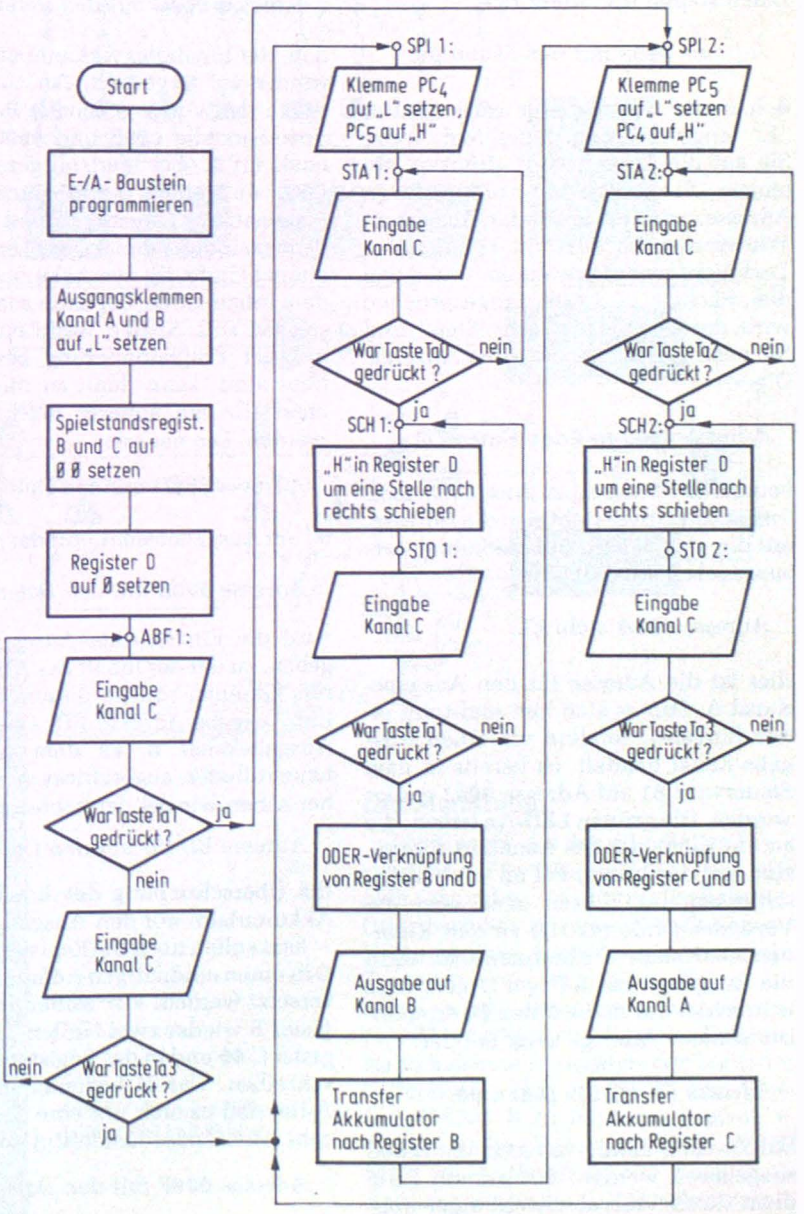


Bild 37. Flußdiagramm des Anwenderprogramms „Elektronisches Spiel ‚Seeschlacht'"

die Taste „Step" zeigt jetzt die

Adresse EF EF mit den Daten 81.

Sowohl bei der Eingabe als auch bei der Ausgabe, die wie vorher erläutert für die Adresse nur 8 Bit benötigen, werden diese 8 Bit sowohl auf den niederwertigen acht Adreßleitungen als auch auf den 8 Bit der höherwertigen Adreßleitungen übertragen. Auf dem Datenbus befindet sich jetzt die 81, die als Daten angezeigt wurden. Nunmehr ist der Ein/Ausgabe-Baustein programmiert; es werden also im Bedienteil die einzelnen Lampen aufleuchten. Ein weiterer Tastendruck auf die Taste „Step" bringt uns zur

Adresse ØØØ7 mit den Daten 3E.

Es sollen wieder 8-Bit-Daten in den Akkumulator geladen werden. Diese Daten stehen nun unter der

Adresse ØØØ8 mit den Daten ØØ,

d. h. in den Akkumulator werden lauter „L" eingeschrieben. Jedes Mal, wenn Sie auf die Taste „Step" drücken, gelangen Sie entweder zur nächsten Adresse, oder zu Ein- oder Ausgaben. Wir werden es jetzt im kommenden Text nicht weiter erwähnen, wenn also die nächste Adresse angesprochen wird, drücken Sie die Taste „Step", und Sie gelangen zu der nächsten Anzeige. Die

Adresse ØØØ9 mit den Daten D3

beinhaltet wieder einen Ausgabebefehl. Im nächsten Byte steht also die Adresse, an die der Inhalt des Akkumulators ausgegeben wird. In der

Adresse ØØØA steht EC,

dies ist die Adresse für den Ausgabe-Kanal A. Daß es sich hier nicht um einen Eingabe-, sondern um einen Ausgabe-Kanal handelt, ist bereits in dem Steuerwort 81 auf Adresse ØØØ4 gesagt worden. Die grünen LED-Anzeigen, die an die Klemmen des Kanals A führen, sind mit dem einen Pol an +5 V angeschlossen und führen dann über die Vorwiderstände (470 Ω) zu den Klemmen des Kanals. Sie brennen also, wenn die Ausgänge auf L-Pegel liegen. Dies wurde hier durch die Daten ØØ erreicht. Die nächste Anzeige zeigt bei der

Adresse EC EC die Daten ØØ.

Ein Zeichen dafür, daß jetzt die Daten ausgegeben werden. In unserem Falle dient dieser Ausgabebefehl einer doppelten Sicherheit; normalerweise sind durch den Reset-Befehl schon alle internen Speicher des Ein/Ausgabe-Bausteines auf L gestellt worden, und die Lampen leuchteten nach dem Ausgabebefehl „D3 EF" auf. Man soll sich aber beim Programmieren angewöhnen, für alle Register und Ausgänge eindeutig definierte Zustände herzustellen. In diesem Falle, wenn also später einmal aus irgendeinem Grunde das Programm geändert wird, braucht man nicht das ganze Programm durchzugehen, um jetzt eventuelle Fehler zu suchen, die durch die Änderung entstanden sind, wenn man hier immer eindeutige Verhältnisse schafft. Die

Adresse ØØØB mit den Daten 3E

beinhaltet wieder einen Ladebefehl für den Akkumulator. Der zu ladende Inhalt steht in der

Adresse ØØØC mit den Daten ØØ,

d. h. der Inhalt des Akkumulators wird wieder auf Ø gestellt. An sich ist er durch den vorhergehenden Befehl auf Speicherstelle ØØØB und ØØØC immer noch auf Ø, aber hier soll gezeigt werden, wie man bei der Programmierung systematisch vorgeht, so daß wir vor dem nachfolgenden Ausgabebefehl erst einmal eindeutig den Akkumulator mit dem Inhalt laden, der dann ausgegeben werden soll. Später, wenn wir bereits mit der Programmierung besser vertraut sind, kann dann an diese Speicherstelle ein anderer Befehl gesetzt werden. Die nächste

Adresse ØØØD mit den Daten D3

ist ein Ausgabebefehl. Auf der nächsten

Adresse ØØØE mit den Daten ED

wird die Ein/Ausgabe-Adressse angegeben, zu der der Inhalt des Akkumulators kommen soll. In diesem Falle die Ein/Ausgabe-Adresse ED. Dies ist der Ausgabekanal B, an dem die roten Leuchtdioden angeschlossen sind. Daher sehen wir bei der nächsten

Adresse ED ED mit den Daten ØØ

die Überschreibung des Inhaltes vom Akkumulator auf den Ausgangskanal.

Jetzt sollen noch die Register B, C und D in einen eindeutigen Anfangszustand versetzt werden. Wir wollen in das Register B wieder zwei Nullen, in das Register C ØØ und in das Register Ø1 H einschreiben. Das H bedeutet in diesem Falle, daß es sich um eine Sedezimalzahl (Hexacode) handelt. Also:

Adresse ØØØF mit den Daten Ø6.

Dies bedeutet, daß in das Register B etwas eingeschrieben wird, welches sich auf der nächsten

Adresse ØØ1Ø mit den Daten ØØ

befindet. In der

Adresse ØØ11 mit den Daten ØØ

steht der Befehl, daß in das Register C Daten eingeschrieben werden sollen, die in der nächsten

Adresse ØØ12 mit ØØ

stehen. Nun soll noch in das Register D die 01 eingeschrieben werden. Der Maschinen-Befehl lautet: „16", welcher auf der

Adresse ØØ13 mit den Daten 16

steht. Die Ø1 steht auf der

Adresse ØØ14 mit 01.

Als nächstes werden die Eingangsklemmen im Kanal C abgefragt. Der Kanal C hat die Adresse EE. Um beim Assemblieren dem Assembler eindeutig zu sagen, daß es sich um eine Sedezimalzahl (Hexacode) handelt, muß vor das E eine Null gesetzt werden. H hinten heißt, daß es sich um eine Sedezimalzahl handelt. Der Eingabebefehl steht nun auf der

Adresse ØØ15 mit den Daten DB.

In der nächsten

Adresse ØØ16 mit den Daten EE

wird die Adresse des Eingabekanales bestimmt. Da der Mikroprozessor 8080 nur den gesamten Kanal C mit seinen 8 Bit abfragen kann, müssen wir jetzt durch einen UND-Befehl das abzufragende Bit herausselektieren. Näheres darüber finden Sie auch in [3]. Der UND-Befehl „ANI" Ø2 H besagt, daß der Inhalt des Akkumulators mit Ø2 UNDiert wird. Das bedeutet, daß auf dem zweitniederwertigen Bit ein H steht, während alle anderen Stellen L führen. Das Ergebnis, das jetzt wieder im Akkumulator steht, ist nur dann ungleich Null, wenn auf dieser Bit-Stelle ein H gewesen ist.

Der Inhalt des Eingabekanales steht im Akkumulator. Wenn wir jetzt keine Taste gedrückt haben, ist also der Inhalt des Akkumulators ØØ, d.h. auf allen Bits stehen „L". Dies sehen wir beim nächsten Schritt, der die Eingabe darstellt, mit der

Adresse EE EE und den Daten ØØ.

Drücken wir jetzt eine der Tasten, so erscheint bei den Daten entweder Ø1, Ø2, Ø3 oder Ø8, je nachdem, welche Taste wir gedrückt haben. *(Fortsetzung folgt)*

Literatur

[3] *Pelka, H.:* Von der Schaltalgebra zum Mikroprozessor. Franzis-Verlag, München.

Horst Pelka

Bei der Beschreibung des Spielprogramms „Seeschlacht" gehen wir hier noch näher auf einige Eigenarten des Mikroprozessors 8080 ein. Ferner wird eine Methode beschrieben, wie die Spielertasten abgefragt werden können.

Praxis mit Mikroprozessoren

11. Teil

Auf der

Adresse ØØ17 mit den Daten E6

steht nun dieser UNDierbefehl. Auf der nächsten

Adresse ØØ18 mit Daten Ø2

ist das Bitmuster, mit welchem der Akkumulator UNDiert werden soll. Bei dem nächsten Befehl „JNZ" (springe, wenn der Inhalt des Akkumulators nicht Null ist) handelt es sich um einen 3-Byte-Befehl. Die nachfolgenden zwei Bytes geben die Adresse an, zu der das Programm springen soll, wenn der Inhalt des Akkumulators ungleich Null war. Dabei ist es eine besondere Eigenart des Mikroprozessors 8080, daß zuerst die zwei niederwertigen Sedezimalstellen und dann die zwei höherwertigen Sedezimalstellen in der Reihenfolge kommen, d.h. in diesem Falle, daß die Adresse ØØ43 als 43ØØ geschrieben wird. In der

Adresse ØØ19 mit Daten C2

steht der eigentliche Sprungbefehl, in der

Adresse ØØ1A die Daten 43,

das ist das niederwertige Byte der neuen Adresse, in der

Adresse ØØ1B die Daten ØØ,

das ist das höherwertige Byte der Sprungadresse. Da wir keine Taste gedrückt hatten, wird also dieser Sprungbefehl ignoriert und als nächstes folgt die

Adresse ØØ1C mit den Daten DB und die

Adresse ØØ1D mit den Daten EE.

Es handelt sich wieder um einen Eingabebefehl des Kanals EE. Diesen Eingabebefehl benötigen wir, da inzwischen der Inhalt des Akkumulators durch den UNDierungsbefehl auf Speicherstelle ØØ17 verändert wurde und wir daher die anderen Klemmen nicht mehr abfragen können. Es folgt also auf der nachfolgenden

Adresse EE EE mit den Daten ØØ

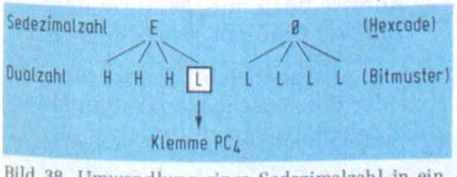


Bild 38. Umwandlung einer Sedezimalzahl in ein Bitmuster

die Überschreibung des Eingabekanales auf den Akkumulator. In diesem Falle eine ØØ. Ein weiterer UNDierungsbefehl auf der

Adresse ØØ1E mit den Daten E6 und

auf der

Adresse ØØ1F die Daten Ø8

selektiert uns die Klemme PC3 heraus, an der Taste Ta3 angeschlossen ist. Auf der nächsten

Adresse ØØ2Ø mit Daten CA

steht der bedingte Sprungbefehl „JZ ØØ15", der besagt, daß zu der Adresse ØØ15 gesprungen werden soll, wenn der Inhalt des Akkumulators Ø ist. Wurde keine Taste gedrückt, so ist der Inhalt Ø und, wie wir nach drei weiteren Schritten sehen werden, wird also das Programm auf die Adresse ØØ15 zurückspringen. Zunächst müssen wir uns aber noch für den Sprungbefehl die Adresse ansehen. Sie steht auf der

Adresse ØØ21 mit Daten 15 und der
Adresse ØØ22 mit Daten ØØ.

Bei einem weiteren Schritt springt also unser Programm auf die Adresse ØØ15 zurück und durchfährt diesen Programmteil, man nennt dies eine Schleife, so lange, bis entweder die Taste Ta1 oder die Taste Ta3 gedrückt wurde. Dieses Abfrageverfahren der Tasten nennt man auch „Polling", im Gegensatz zum „Interrupt", den wir bei einer späteren Folge kennenlernen werden. Wir durchfahren jetzt die Schleife noch einmal, drücken aber während der Eingabe nach Adresse ØØ1D die Taste Ta3, dies wird während der Eingabe auf dem Datenbus mit Ø8 angezeigt.

Jetzt wird also nach der Adresse ØØ22 die

Adresse ØØ23 mit den Daten 3E

erscheinen. Der Maschinencode „3E" bedeutet wieder, daß in den Akkumulator Daten eingeschrieben werden sollen; in diesem Falle handelt es sich um die Daten der

Adresse ØØ24, das ist EØ.

Bei der Sedezimalzahl EØ (Hexcode) handelt es sich um ein Bitmuster (siehe *Bild 38*), welches, auf den Ausgang C gegeben, an der Klemme P_{C4} L-Pegel hervorruft. Durch den nachfolgenden Ausgabebefehl auf

Adresse ØØ25 mit Daten D3

und nachfolgender

Adresse ØØ26 mit Daten EE

wird das Bitmuster EØ auf den Ausgang mit der Adresse EE (Kanal C) gegeben. Wir sehen also bei unserer Anzeige als nächstes auf der

Adresse EE EE die Daten EØ.

Dies bedeutet, daß über den Datenbus die EØ vom Akkumulator auf den Ausgang des Kanals C gegeben wird. In diesem Falle sind nur die vier höherwertigen Bits als Ausgabe programmiert.

Als nächstes erscheinen bei

Adresse ØØ27 die Daten DB

und bei der nachfolgenden

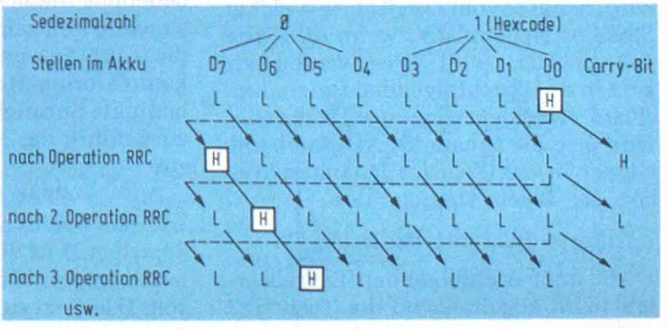
Adresse ØØ28 die Daten EE.

Es handelt sich also um einen Eingabebefehl des Kanals C, welcher die Adresse EE hat. Diese Eingabe sehen wir auch jetzt als nächstes auf unserer Anzeige. Mit

Adresse EE EE und Daten EØ,

wenn keine der Tasten gedrückt ist.

Bild 39. ► Beschreibung des Rotierbefehles „RRC" und seine Auswirkung auf das Bitmuster Ø1H nach jeder Operation



Auch hier können wir wieder durch Drücken der unterschiedlichen Tasten auf dem Datenbus E1, E2, E4 und E8 erzielen. Die nachfolgende

Adresse ØØ29 mit Daten E6 und
Adresse ØØ2A mit Daten Ø1

bewirkt eine UNDierung mit der Sedezimalzahl Ø1. Es ist die Selektierung der Klemme P_{CO}, an der die Taste TAO angeschlossen ist. Der nachfolgende bedingte Sprungbefehl „JZ" führt das Programm wieder zur Adresse ØØ27 zurück, wenn die Taste nicht gedrückt war. Der Sprungbefehl lautet: Springe zur Adresse ØØ27, wenn der Inhalt des Akkumulators Ø ist. Wenn nämlich die Taste nicht gedrückt war, ist nach dem UNDieren im Akkumulator der Inhalt Ø. Wir sehen diesen Sprungbefehl auf

Adresse ØØ2B mit den Daten CA,
Adresse ØØ2C mit den Daten 27 und
Adresse ØØ2D mit den Daten ØØ.

Es soll noch einmal daran erinnert werden, daß bei den Sprungbefehlen die Adressen zuerst mit dem niederwertigen Byte und dann mit dem höherwertigen Byte angegeben werden. War also keine Taste gedrückt, wird als nächstes wieder die

Adresse ØØ27 mit den Daten DB

erscheinen. Diese Schleife wird so lange fortgesetzt, bis die Taste Ta0 gedrückt worden ist. Jetzt können wir die

Adresse ØØ2E mit den Daten 7A

erreichen. Der Maschinencode 7A heißt, daß der Inhalt des Registers D in den Akkumulator geschoben wird. Wir hatten bei den Speicherstellen ØØ13 und ØØ14 in das Register D und Ø1 eingeschrieben. Diese gelangt jetzt in den Akkumulator und wird mit dem nachfolgenden Befehl „RRC" um eine Stelle nach rechts geschoben. Dieser Befehl steht in der

Adresse ØØ2F mit den Daten ØF.

Anschließend wird der Inhalt des Akkumulators wieder in das Register D zurückgeschoben. Da in der nachfolgenden Schleife jedesmal beim Überfahren der Adresse ØØ27 das H (*Bild 39*) um eine Stelle nach rechts geschoben wird, hängt es vom Zufall ab, in welcher Stelle es stehenbleibt, wenn die Taste Ta1 gedrückt wird. Dies werden wir jetzt in dem nachfolgenden Programmablauf verfolgen können. Als nächstes wird also der Inhalt des Akkumulators wieder in das Register D eingeschrieben. Der Befehl steht auf der

Adresse ØØ3Ø mit den Daten 57.

Mit dem nachfolgenden Eingabebefehl wird abgefragt, ob die Taste Ta1

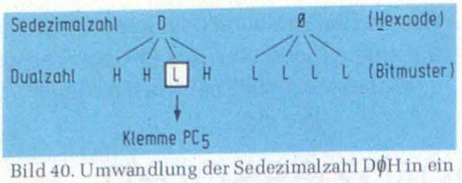


Bild 40. Umwandlung der Sedezimalzahl DØH in ein Bitmuster

gedrückt ist. Der Eingabebefehl steht in der

Adresse ØØ31 mit den Daten DB und
Adresse ØØ32 mit den Daten EE.

Wir sehen also auf der Adresse EE EE entweder die Daten EØ, wenn keine Taste gedrückt ist, oder die Daten E2, wenn Taste Ta1 gedrückt wurde.

Der Befehl „ANI" Ø2 H macht wieder die Selektierung der Taste Ta1. Er steht auf der

Adresse ØØ33 mit Daten E6 und
Adresse ØØ34 mit Daten Ø2.

Die nachfolgenden sechs Adressen beinhalten eine Hilfskonstruktion, um die Adresse ØØ38 für einen späteren „Interrupt" freizuhalten. Wir geben unserem Programm den Befehl, daß die Fortsetzung bei der Adresse ØØ3B erfolgt. Dies wird durch den unbedingten Sprungbefehl „JMP" bewirkt, der auf der

Adresse ØØ35 mit den Daten C3,
Adresse ØØ36 mit den Daten 3B und
Adresse ØØ37 mit den Daten ØØ steht.

Die

Adresse ØØ38 mit Daten ØØ,
Adresse ØØ39 mit Daten ØØ und
Adresse ØØ3A mit Daten ØØ

sind für einen 3 Byte-Sprungbefehl für die „Interrupt"-Funktion freigehalten. Als nächstes kommt die

Adresse ØØ3B mit Daten CA, die
Adresse ØØ3C mit Daten 2E und
Adresse ØØ3D mit Daten ØØ.

Dies ist ein bedingter Sprung zur Adresse ØØ2E, nämlich dann, wenn der Akkumulatorinhalt Ø ist, also die Taste Ta1 nicht gedrückt wurde. Die Schleife läuft so lange, bis die Taste Ta1 gedrückt wird, dabei wird jeweils bei einem Schleifendurchlauf das H-Bit um eine Stelle nach rechts verschoben, so daß es beim Drücken der Taste Ta1 an beliebiger Stelle stehen kann, da dieser Vorgang sehr schnell abläuft. Wird nun die Taste Ta1 gedrückt, so ist der Akkumulatorinhalt nicht mehr Ø, und der bedingte Sprungbefehl „JZ" wird nicht ausgeführt; das Programm läuft weiter zur

Adresse ØØ3E mit den Daten 7A.

Dieser Befehl sagt, daß der Inhalt des Registers D in den Akkumulator übernommen werden soll. Inhalt des Registers D ist jetzt ein Bit-Muster, in dem an irgendeiner Stelle, die sich durch den Zeitpunkt des Drückens der Taste Ta1 ergeben hat, ein H steht. Das H kann aber auch zufällig im Carry-Bit sein, so daß der Inhalt des Akkumulators L ist. Durch eine ODER-Verknüpfung mit dem Register B wird jetzt an der Stelle, an der ein H im Akkumulator steht, dieses auf das Register B übertragen. Das Ergebnis dieser ODERierung steht im Akkumulator. Auf unsere Leuchtdioden bezogen, heißt dies, daß an der Stelle, an der jetzt ein H übertragen wurde, die betreffende, an den Kanal B angeschlossene Leuchtdiode erlischt, denn der andere Pol ist an + 5 V angeschlossen; es besteht also jetzt Potentialgleichheit zwischen den beiden Polen, so daß die Leuchtdiode nicht mehr leuchten kann. Das Register B enthält also je nach dem Spielstand die gelöschten Leuchtdioden, weitere Treffer können weitere Leuchtdioden löschen; am Ende des Spiels enthält das Register B lauter H. Diese ODERierung wird durch den Befehl auf der

Adresse ØØ3F mit Daten BØ

ausgeführt. Das Ergebnis steht, wie gesagt, im Akkumulator. Der nächste Befehl ist ein Ausgabebefehl auf

Adresse ØØ4Ø mit Daten D3 und
Adresse ØØ41 mit Daten ED.

Bei ED handelt es sich wieder um die Adresse des Kanales B. Als nächstes folgt also die Ausgabe mit

Adresse ED ED und den Daten,

die zufällig sich durch die Lage des H ergeben haben.

Mit dem Befehl auf

Adresse ØØ42 und Daten 47

wird der Spielstand, der jetzt im Register stand, wieder auf das Register B übertragen. Jetzt muß noch die gelbe Lampe des Spielers gelöscht und die gelbe Lampe des Gegenspielers eingeschaltet werden. Zuerst wird also ein Muster in den Akkumulator gebracht, welches diese Umschaltung vollzieht. Es ist das Muster ØDØH (*Bild 40*). Der Befehl zur Einschreibung des Musters in den Akkumulator steht in der

Adresse ØØ43 mit Daten 3E und
Adresse ØØ44 mit den Daten DØ.

Der nächste Befehl auf

Adresse ØØ45 mit Daten D3 und
Adresse ØØ46 mit Daten EE

ist ein Ausgabebefehl. Es wird also jetzt wieder der Ausgabekanal doppelt auf der Adreßanzeige zu finden sein:

Adresse EE EE mit Daten DØ.

(Fortsetzung folgt)

Horst Pelka

Im vorliegenden 12. Teil schließt der Verfasser die Beschreibung des Programms für das Seeschlacht-Spiel ab. In der nächsten Folge werden weitere Spielprogramme erstellt und beschrieben.

Praxis mit Mikroprozessoren

12. Teil

Jetzt haben wir auch beobachtet, daß sich die gelbe Lampe von dem einen Spieler gelöscht hat und die gelbe Lampe des anderen aufleuchtet. Wir gelangen zu einer weiteren Schleife, die im wesentlichen die Abfrage ist, ob die Taste Ta2 gedrückt ist. Mit dem Eingabebefehl „IN" auf

Adresse ØØ47 mit den Daten DB und
Adresse ØØ48 mit den Daten EE

wird der Kanal C eingegeben. Kanal C hat die Adresse EE. Unsere Anzeige zeigt also auf

Adresse EE EE die Daten DØ,

wenn keine Taste gedrückt ist. Wir drücken aber die Taste Ta2, es erscheinen jetzt auf dem Datenbus die Daten D4. Auf

Adresse ØØ49 stehen die Daten E6

und

Adresse ØØ4A die Daten Ø4.

E6 Ø4 ist der UNDierungsbefehl „ANI Ø4H", d.h. aus dem Kanal C wird die Klemme P_{C2} abgefragt. War die Taste gedrückt, ist der Inhalt des Akkumulators nicht Null; der nachfolgende Sprungbefehl „JZ" (Springe, wenn der Inhalt des Akkumulators Null ist) wird nicht ausgeführt. Dieser Befehl steht auf

Adresse ØØ4B mit Daten CA,
Adresse ØØ4C mit Daten 47 und
Adresse ØØ4D mit Daten ØØ.

Da die Taste gedrückt war, geht das Programm auf die nächste

Adresse ØØ4E mit Daten 7A über.

Dieser Befehl ist wieder ein Transferbefehl. Der Inhalt des Registers D wird in den Akkumulator transferiert. Im nächsten Befehl auf

Adresse ØØ4F mit Daten ØF

werden sämtliche Stellen des Akkumulators um eine Stelle nach rechts gerückt, d.h. auch das irgendwo sich befindende H, welches dann später die Lampen löscht. Der nächste Befehl auf

Adresse ØØ5Ø mit Daten 57

ist ein Transferbefehl, welcher den Inhalt des Akkumulators wieder in das Register D zurückschreibt. Die Schleife geht über die Adressen ØØ4E bis 57. Als nächstes folgt ein Eingabebefehl des Kanals C, welcher auf

Adresse ØØ51 mit Daten DB und
Adresse ØØ52 mit Daten EE steht.

Wir haben als nächsten Schritt den Eingabevorgang. Es stehen also auf der Anzeige der

Adresse EE EE die Daten DØ,

wenn keine der Tasten gedrückt ist. In diesem Falle springt also das Programm von der Adresse ØØ57 auf ØØ4E zurück, bis die Taste Ta3 gedrückt ist. Wir drücken also jetzt die Taste Ta3. Auf dem Datenbus stehen dann die Daten D8. Mit dem nächsten UNDierungsbefehl wird die Klemme P_{C3}, an der die Taste Ta3 angeschlossen ist, selektiert. Dieser Befehl steht auf

Adresse ØØ53 mit Daten E6 und
Adresse ØØ54 mit Daten Ø8.

Nachfolgend der bedingte Sprungbefehl „JZ", wenn der Akkumulator den Inhalt Null gehabt hätte. Er ist auf

Adresse ØØ55 mit Daten CA,
Adresse ØØ56 mit Daten E4 und
Adresse ØØ57 mit Daten ØØ

untergebracht. Der nächste Befehl ist wieder ein Transferbefehl. Er steht auf

Adresse ØØ58 mit Daten 7A.

Der Inhalt des Registers D wird in den Akkumulator übertragen. Es handelt sich um ein einzelnes H, welches an irgendeiner Stelle steht. Die anderen Stellen haben alle L. Der ODERierungs-Befehl „ORA C" bringt dieses H in das Spielstandsregister C ein, d.h. am Ende des Spieles, wenn alle Lampen gelöscht sind, haben alle Stellen H. Der ODERierungsbefehl steht auf

Adresse ØØ59 mit Daten B1.

Da die entsprechenden Lampen jetzt auch gelöscht werden sollen, folgt also als nächstes ein Ausgabebefehl. Das Ergebnis dieser ODERierung steht ja im Akkumulator. Der Ausgabebefehl auf

Adresse ØØ5A mit Daten D3 und
Adresse ØØ5B mit Daten EC

gibt den Akkumulator-Inhalt auf den Kanal A aus. Folglich zeigt unsere Anzeige auf

Adresse EC EC z. B. die Daten 8Ø.

Das höchstwertige Bit hat also H angenommen. Die entsprechende Lampe ist gelöscht. Die Daten hier bei der Ausgabe wechseln je nach Spielstand. Wenn alle Stellen H-Signal haben, wird also an dieser Stelle FF ausgegeben. Mit dem Befehl auf

Adresse ØØ5C und Daten 4F

„MOV C, A" wird jetzt noch der Inhalt des Akkumulators in das Spielstandsregister C übertragen. Nun soll als nächstes wieder der Spieler 1 drankommen. Wir haben hier einen unbedingten Sprungbefehl nach Adresse ØØ23. Dieser Sprungbefehl „JMP" steht auf

Adresse ØØ5D mit Daten C3,
Adresse ØØ5E mit Daten 23 und
Adresse ØØ5F mit Daten ØØ.

Als nächsten Schritt springen wir also wieder auf

Adresse ØØ23 mit Daten 3E.

Das Spiel kann so lange fortgesetzt werden, bis bei einem Spieler alle Lampen gelöscht sind. Selbstverständlich wird jetzt das Spiel nicht in Einzelschritten durchgeführt, sondern der Schalter „Einzelschritte" wird ausgeschaltet, so daß der Mikroprozessor mit seinem normalen Takt arbeiten kann. Den weiteren Ablauf können Sie selbstverständlich auch in Einzelschritten verfolgen, um die Funktionsweise des Mikroprozessors schrittweise zu verfolgen.

Anwendung der Zusatzschaltung zur Anzeige der Daten von einer vorgewählten Adresse

Wir können jetzt auch im Betrieb von einer beliebigen Adresse die Daten anzeigen. Dazu ist es aber wichtig, daß der Mikroprozessor mindestens einmal diese Adresse durchläuft, da sonst die Daten nicht in die Pufferspeicher der 7-Segment-Dekoder eingeschrieben werden können. Um festzustellen, in welcher Schleife der Mikroprozessor gerade arbeitet, stellen wir zunächst den Schalter „Einzelschritte" auf „Single-step" und drücken einmal die Taste „Single-step". Jetzt wird beispielsweise die Adresse ØØ2C mit Daten 27 angezeigt. Gehen wir jetzt auf unser Programm (Bild 29 der Folge 5) so

sehen wir, daß es sich um die Schleife zwischen den Adressen ØØ27 und ØØ2D handelt.
Wir stellen jetzt also mit den Adreßschaltern am Modul für direkten Speicherzugriff oder mit den Adreßschaltern für die Zusatzeinrichtung, wenn Sie dort getrennte Adreßschalter verwendet hatten, beispielsweise die Adresse 0029 ein. Das bedeutet, daß die Adreßschalter A0, A3 und A5 auf H, alle anderen Adreßschalter auf L stehen müssen. Es wird jetzt die Adresse 0029 erscheinen mit den Daten E6. Ein Blick in's Programm sagt uns, daß es sich hier um die richtigen Daten handelt.

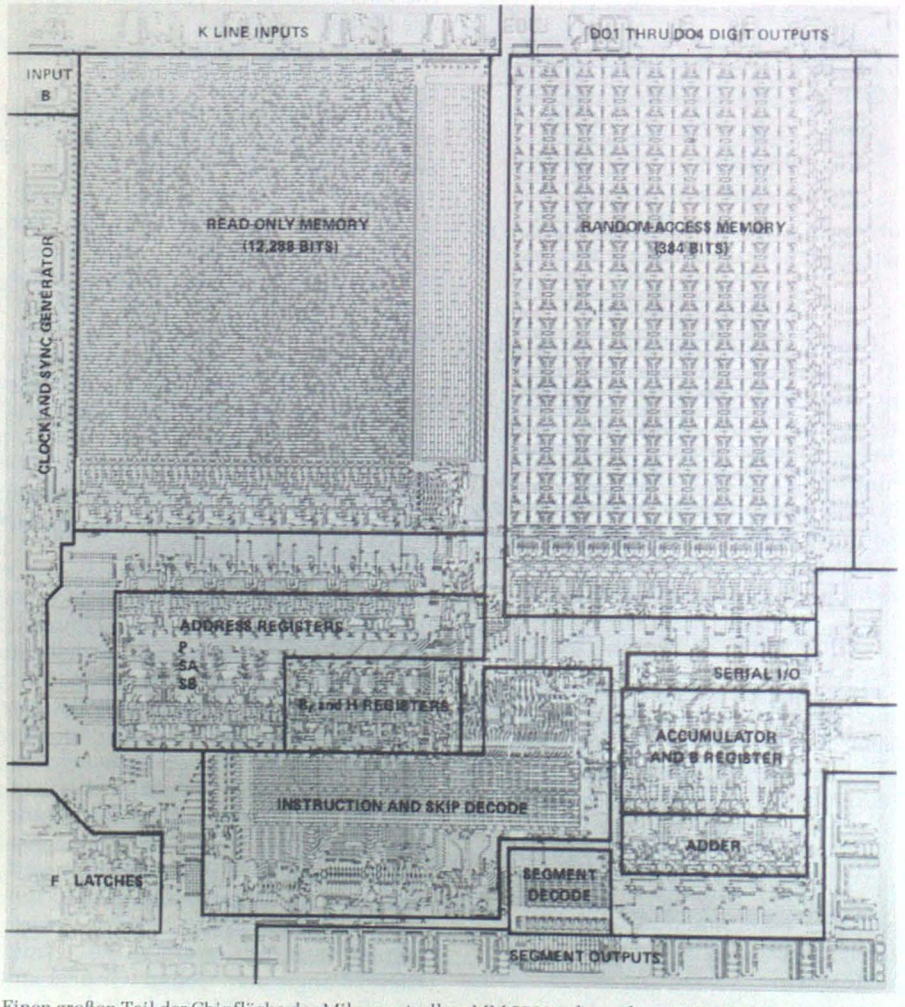
Jetzt möchten wir uns einen anderen Bereich ansehen, beispielsweise die Adresse ØØØD, die aber jetzt nicht in der augenblicklichen Schleife liegt. Bei der Adresse ØØØD stehen die Adreßschalter AØ, A2 und A3 auf H, alle anderen Adreßschalter auf L. Um in diesen Bereich zu kommen, müssen wir die Taste „Reset" drücken und jetzt erscheint auch die richtige Adresse mit den Daten D3. Ein Vergleich mit unserem Anwenderprogramm in Bild 29 sagt uns, daß dies die richtigen Daten sind.

Umgekehrt herum können wir natürlich auch damit feststellen, in welchem Schleifenbereich der Mikroprozessor arbeitet, nämlich indem wir zunächst alle Adreßschalter auf L stellen, die Taste „Reset" drücken, dann erscheinen auf der Adreßanzeige und der Datenanzeige lauter Nullen. Jetzt suchen wir mit dem Adreßschalter so lange, bis sich die Anzeige verändert. Beispielsweise stellen wir zunächst den Adreßschalter A4 auf H. Es erfolgt keine Änderung. Wir stellen den Adreßschalter A5 auf H. Es erscheinen auf der Adresse ØØ2Ø die Daten CA. Jetzt wissen wir, daß die Schleife mindestens die Adresse ØØ2Ø überfährt, und können anhand des Anwenderprogramms uns die entsprechende Schleife heraussuchen.

Für die Leser, die bisher mit Datentechnik noch nicht in Berührung kamen, ist dieses natürlich jetzt ein ganz neues Gebiet, und es gehört schon einige Mühe dazu, sich dort hineinzufinden. Es ist sinnvoll, sich noch einmal das Programm anhand des Flußdiagrammes und des tabellarisch ausgedruckten Anwenderprogrammes in Einzelschritten anzusehen. In den nächsten Folgen wird durch weitere Anwendungsbeispiele die Materie vertieft werden. *(Fortsetzung folgt)*

Neue IS-Familie für Mikrocontroller-Anwendungen

Unter der Bezeichnung „Calculator oriented processors" (COPS) brachte National Semiconductor eine Reihe neuer IS heraus, die sich nur in der Speicherkapazität und im Ein- und Ausgabeformat unterscheiden, aber auf einem Chip bereits alle für ein kleines Computersystem notwendigen Elemente enthalten: eine Rechen- und Steuereinheit, ein ROM (Read-only Memory), ein RAM (Random Access Memory) und geeignete I/O-(Input/Output-)Schaltungen. Diese Entwicklung schließt die Lücke zwischen leistungsfähigen Mikroprozessor-Systemen und kundenspezifischen IS; sie ermöglicht auch bei mittleren Stückzahlen noch eine günstige Preisgestaltung. Als Anwendungsbereiche kommen Tischrechner, Registrierkassen oder elektronische Waagen ebenso in Frage wie etwa industrielle Steuerungen kleiner bis mittlerer Größe. Kundenspezifisch ist hier nur noch die Programmierung des ROM, die durch den IS-Hersteller erfolgt. Dabei wird von National ein Zeitraum von nur etwa 16 Wochen von der Problemstellung des Kunden bis zur Lieferung erster Prototypen angegeben, was sich bei kundenspezifischen IS praktisch nicht erreichen läßt.



Einen großen Teil der Chipfläche des Mikrocontrollers MM 5799 nehmen bereits die Speicherbereiche ein

Der Chip des Typs MM 5799 enthält z. B. 1536 8-bit-Befehle im ROM, und das RAM ist für eine Kapazität von 96 BCD-Stellen von je 4 bit ausgelegt. Diese beiden Schaltungsteile nehmen den größten Teil der Chipfläche ein (*Bild*); weitere Elemente sind ein Adressenregister, ein Akkumulator, ein Register für Zwischenergebnisse, ein Übertragsspeicher und ein programmierbarer Logikteil zur Umsetzung von BCD-Eingangssignalen in Siebensegment-Ausgangssignale. Dank interner Pufferschaltungen lassen sich LED-Anzeigen direkt ansteuern. Zusammen mit einem zusätzlichen 1024-bit-RAM, z. B. MM 57126, läßt sich auf einfache Weise ein programmierbarer Rechner aufbauen.

Horst Pelka

Zahlreiche Anfragen aus Leserkreisen veranlassen den Autor, zu dem Problem einer käuflichen „Hardware" für den Aufbau des in den vergangenen Folgen beschriebenen Mikrocomputers Stellung zu nehmen. Offensichtlich besteht häufiger als erwartet der Wunsch, geprüfte „Hardware" fertig zu beziehen und nur die „Software" selbst zu erstellen.

Praxis mit Mikroprozessoren

13. Teil

Kompatibiltät

Kompatibilität mit anderen Systemen

Wie bereits im 1. Teil* gesagt wurde, ist die Steckerbelegung der Rückwandverdrahtung identisch mit dem SMP 80-System der Fa. Siemens. Die Aufteilung der Modul-Funktionen

Mikroprozessor
Speicher
E/A-Bausteine

stimmt damit ebenfalls überein (*Bild 41*). Das SMP 80-System ist für eine universelle professionelle Anwendung entwickelt worden. Es enthält daher wesentlich mehr Bausteine, als das hier in der FUNKSCHAU beschriebene, auf sparsamen Aufbau ausgelegte System. Unter Beachtung einiger kleinerer Anpassungen ist das SMP 80-System jedoch verwendbar, wenn auf die batteriegepufferten RAMs verzichtet wird. Den Anzeige-Modul und Modul für direkten Speicherzugriff gibt es in der SMP 80-Reihe nicht.

Eindeutig muß festgestellt werden, daß die FUNKSCHAU-Module nicht in einem SMP 80-System verwendet werden können, da alle SMP 80-Baugruppen mit Daten- und Adreßbuspuffern ausgerüstet sind und daher ohne zusätzliche Maßnahmen erweiterbar sind, während bei einer Erweiterung unseres Systems aus Kostengründen erst von Fall zu Fall Bausteine ergänzt werden (siehe Adreßbuspuffer im Mikroprozessor-Modul, Erweiterung der Speicherbestückung im Speicher-Modul oder E/A-Bausteine im Ein-Ausgabe-Modul).

Zentralbaugruppe des SMP 80-Systems

Die Zentralbaugruppe B 88080-A22-E1** ist ohne Änderung einsetzbar. Es fehlt jedoch an diesem Modul ein RESET-Ausgang, welcher beim Funkschau-Modul am Anschluß 15c (ZF1) als Zusatzfunktion herausgeführt ist. Die Siemens-Baugruppe hat dafür an Anschluß 6 a das $\overline{\text{RESET}}$-Signal. Werden also Funkschau-Module in Zusammenspiel mit dem Modul E1 benutzt, muß entweder das RESET-Signal mit einem Inverter aus dem $\overline{\text{RESET}}$ gewonnen werden, oder eine Drahtverbindung zwischen dem Anschluß 15c der Sitftleiste und Anschluß 1 des Bausteines SAB 8224 hergestellt werden.

Die Siemens-Baugruppe enthält zusätzlich eine Serien-Schnittstelle mit dem Baustein SAB 8251. Diese Serien-

FUNKSCHAU-System	Anzeige-Modul (Teil 1/2)	Modul für direk. Speicherzugriff (Teil 2/3)	Speicher-Modul (Teil 4-5)	Mikroprozessor-Modul (Teil 6)	Eingabe-/Ausgabe-Modul (Teil 7)	Netzteil mit Hilfsschaltung zur Datensperre (Teil 3/4)
SMP 80-System	Nicht vorhanden	Nicht vorhanden	Speicherbaugruppe B 88080-A2-E120	Zentralbaugruppe B 88080-A22-E1	parallele E/A-Baugruppe B 88080-A2-E200	Netzteil E901 + E902 oder E420

Bild 41. Vergleich der Baugruppen beim FUNKSCHAU-System mit den nahezu kompatiblen Baugruppen des Siemens SMP 80-Systems

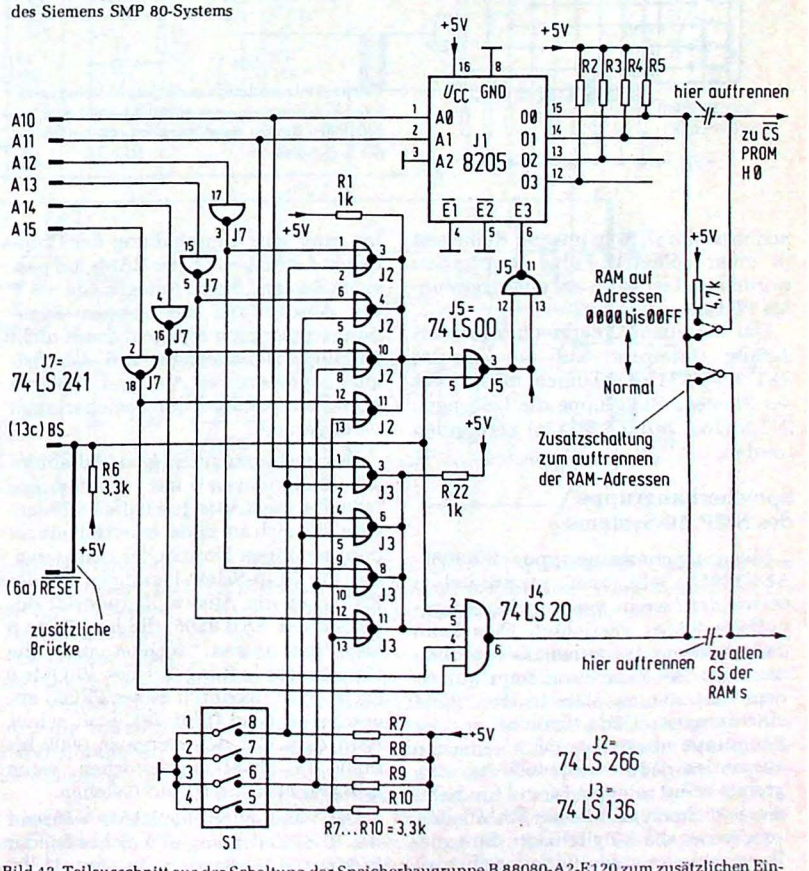


Bild 42. Teilausschnitt aus der Schaltung der Speicherbaugruppe B 88080-A2-E120 zum zusätzlichen Einbau eines Adressenumschalters für die RAM-Adressen und die Sperrung der RAMs während des RESET-Vorganges

* FUNKSCHAU 1977, Heft 15, Seite 675

** Anfragen über Preise und Bezugsmöglichkeiten richten Sie bitte an die nächste Zweigniederlassung der Siemens AG, Abt. VB.

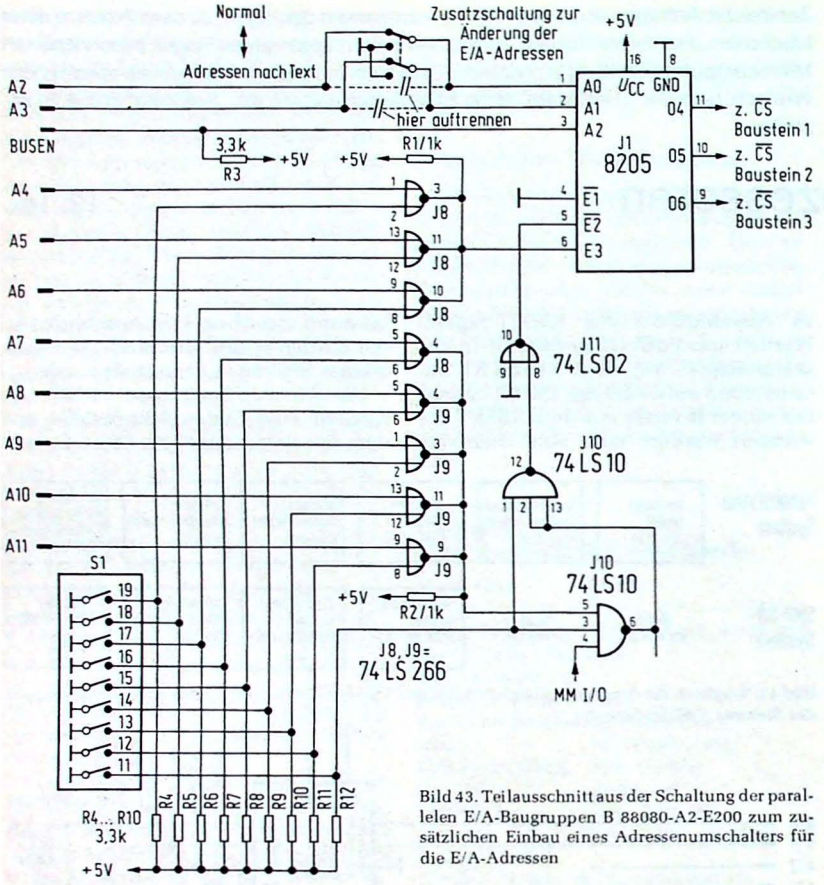


Bild 43. Teilausschnitt aus der Schaltung der parallelen E/A-Baugruppen B 88080-A2-E200 zum zusätzlichen Einbau eines Adressenumschalters für die E/A-Adressen

Schnittstelle wird in unserer Reihe erst in einer späteren Folge besprochen werden und ist dann auf einer getrennten Platine aufgebaut.

Um bei einer Unterbrechungsanforderung (Interrupt) auf die Adresse RST 7 = 38 H zu gelangen, müssen bei der Siemens-Baugruppe die Leitungen INTA (23a) mit ZF 5 (23c) verbunden werden.

Speicherbaugruppe des SMP 80-Systems

Die Speicherbaugruppe B 88080-A2-E120** läßt sich grundsätzlich verwenden, wenn man auf batteriegepufferte RAMs verzichtet. Dies kann dann erfolgen, wenn man Gelegenheit hat, das Anwender-Programm auf einem Entwicklungsplatz in ein PROM einzubringen. Dann braucht an der Baugruppe ebenfalls nichts verändert zu werden. Will man jedoch das Programm selbst mit dem Modul für direkten Speicherzugriff in ein RAM laden, so gäbe es die Möglichkeit, die ganze Baugruppe an einen Batteriepuffer anzuschließen oder auch nur die Speisespannung für die RAMs. In beiden Fällen muß eine Umschaltung der Chip-Selekt-Leitungen $\overline{CE}$ der RAMs auf passiven Zustand durch Anlegen von +5 V *vor* Abschaltung der übrigen Spannungsversorgung erfolgen, damit nicht zufällige Steuersignale und Adreßsignal während des Ab- und späteren Einschaltvorganges den Speicherinhalt verändern.

Des weiteren muß eine Adreßumschaltmöglichkeit mit Startadresse 0000 für die RAMs geschaffen werden. Dies läßt sich am einfachsten mit einem doppelpoligen Umschalter realisieren, der die Chip-Selekt-Leitungen $\overline{CE}$ der RAMs an die Auswahlleitung 00 des Bausteines SAB 8205, die für PROM 0 bestimmt waren, schaltet und die Chip-Selekt-Leitung $\overline{CS}$ des PROMs 0 nach +5 V, wodurch dieses PROM abgeschaltet wird (*Bild 42*). Statt seiner wird dann bei den Adressen 0000 bis 00FF das RAM angesprochen, wenn alle Schalter von S1 auf 0 stehen.

Das Abschalten der RAMs während des RESET-Vorganges, welches bei der FUNKSCHAU-Version in der Hilfsschaltung zur Datensperre erfolgt, kann beim SMP 80-System durch Verbinden der $\overline{RESET}$-Leitung (6a) mit BS (13c) (Nur *an diesem einen Modul verbinden!!*) erfolgen.

Parallele Ein-/Ausgabebaugruppe des SMP 80-Systems

Bei der Baugruppe B 88080-A2-E200** werden die Leitungen ZF0 bis ZF5 für andere Zusatzfunktionen benutzt. Da bei der Rückwandverdrahtung die Leitungen ZF0 bis ZF8 nicht durchverbunden sind, sondern mit Hand verdrahtet werden müssen, muß darauf geachtet werden, daß der Modul immer an einen Platz gesteckt wird, bei dem die Zusatzfunktionsleitungen nicht beschaltet sind.

Am Frontstecker müssen die +5-V-Leitung (22a) $\overline{INT}$ (22b) und $\overline{RESIN}$ (22c) mit Handverdrahtung ergänzt werden. Der elektronische Umschalter $\overline{MMI/O}$ (Memory mapped Input/Output), Ein-Ausgabe über Speicheradressen, Anschluß 4c, muß an +5 V gelegt werden, damit die Ein-Ausgabe normal abläuft.

Die erste Stelle der Sedezimalzahl für die E/A-Adresse läßt sich mit dem Adreßschalter S 1 frei wählen. Die *Tabelle 1* zeigt die Adressen für die 3 E/A-Bausteine beim SMP 80-System im Vergleich zum FUNKSCHAU-Modul. Eine Angleichung ist entweder durch Änderung der E/A-Adressen im Anwenderprogramm nach *Tabelle 2* oder durch Einbau eines Umschalters nach *Bild 43* möglich. *(Fortsetzung folgt)*

Tabelle 1

SMP 80-E/A-Baugruppe	FUNKSCHAU E/A-Modul
Baustein 1: X0 bis X3	EC bis EF
Baustein 2: X4 bis X7	DC bis DF
Baustein 3: X8 bis XA	BC bis BF

Für X kann jede Sedezimalzahl zwischen 0 und F gewählt werden. Sie wird am Adreßschalter S 1 eingestellt.

Tabelle 2***

0005	D3E3	OUT	0E3H
0009	D3E0	OUT	0E0H
000D	D3E1	OUT	0E1H
0015	DBE2	IN	0E2H
001C	DBE2	IN	0E2H
0025	D3E2	OUT	0E2H
0027	DBE2	IN	0E2H
0031	DBE2	IN	0E2H
0040	D3E1	OUT	0E1H
0045	D3E2	OUT	0E2H
0047	DBE2	IN	0E2H
0051	DBE2	IN	0E2H
005A	D3E0E	OUT	E0H

*** Nur bei FUNKSCHAU-Moduln.

Nachdem wir in den vorhergehenden Folgen ein einfaches Programm für das elektronische Spiel „Seeschlacht" kennengelernt haben, wollen wir uns heute mit zwei neuen Begriffen beschäftigen. Dies sind Unterbrechungen (Interrupts) und Unterprogramme (Subroutines).

Horst Pelka:

Praxis mit Mikroprozessoren

14. Teil

Bei der Softwareänderung sind folgende neuen Adressen zu verwenden:

Adresse im FUNKSCHAU-E/A-Modul	Neue Adresse für ungeänderte Siemens-E/A-Baugruppe
Baustein 1	
E C	X Ø
E D	X 1
E E	X 2
E F	X 3
Baustein 2	
D C	X 4
D D	X 5
D E	X 6
D F	X 7
Baustein 3	
B C	X 8
B D	X 9
B E	X A
B F	X B

Für X kann jede beliebige Sedezimalzahl von Ø bis F eingesetzt werden. Die Zahl muß nur mit der Einstellung am Adreßschalter übereinstimmen.

Soll das Originalprogramm verwendet und zunächst nur ein E/A-Baustein benutzt werden, so ist der einfachste Weg ein zweipoliger Umschalter, der die A_{07} und A_{F}-Eingänge des Bausteines SAB 8205 wahlweise auf Masse legt (Bild 43). Dann wird der Baustein von folgenden Adressen gleichzeitig angesprochen:

Y Ø	Y 4	Y 8	Y C
Y 1	Y 5	Y 9	Y D
Y 2	Y 6	Y A	Y E
Y 3	Y 7	Y B	Y F

Für Y ist die am Adreßschalter S 1 eingestellte Sedezimalzahl einzusetzen. Wird dieser beispielsweise auf „E" gestellt, so finden wir in der rechten Spalte die in unserem Originalprogramm verwendeten E/A-Adressen.

Netzteil

Als Netzteil können entweder die Typen B 88080-A2-E 901 + E 902 (beide erforderlich) oder der Typ B 88080-A2-E 420** verwendet werden. Die in der FUNKSCHAU-Serie Teil 4 beschriebene Hilfsschaltung zur Datensperre ist nicht fertig erhältlich.

Rückwandverdrahtung

Als Rückwandverdrahtung kann, wie bereits in Folge 4 erwähnt, die Bauform B 88080-A3-S400** fertig bezogen werden. Es müssen mit Hand verdrahtet (verbunden) werden (*Bild 44*):

ZF0: Hilfsschaltung zur Datensperre (in FUNKSCHAU-
(13c Netzteil) mit Speicher-Modul***

ZF1(RESET):
(15c)
Mikroprozessor-Modul mit Speicher-Modul und Ein-/Ausgabe-Modul und Hilfsschaltung zur Datensperre***

ZF2 (Strobe-Impuls):
(17c)
Anzeige-Modul mit Modul für direkten Speicherzugriff (einschl. Zusatzschaltung zur Anzeige der Daten von einer vorgewählten Adresse im Betrieb)
Ferner ist es sinnvoll, die Masse und +5-V-Leitungen mit Draht von > 1,5 mm Ø zu verstärken.

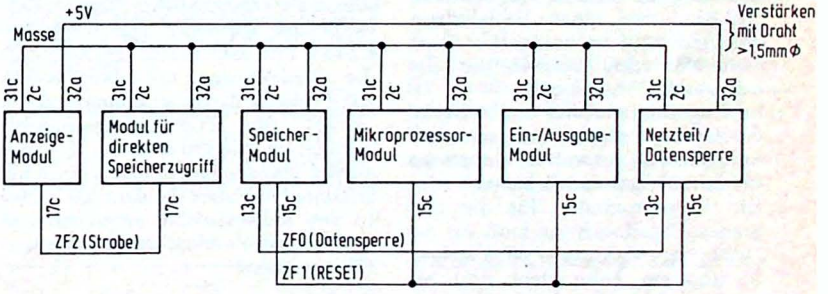


Bild 44. Mit Handverdrahtung auszuführende Verbindungen bei Verwendung einer Rückwandverdrahtung B 88080-A3-S400

Programmunterbrechungen

Beim 8080- und 8085-System ist die Möglichkeit für Programmunterbrechungen (Interrupts) gegeben. Diese Unterbrechungsmöglichkeit kann man vielseitig ausnutzen, insbesondere dann, wenn ein Programm mit höherer Priorität abgearbeitet werden, d. h. zwischen ein laufendes Programm eingeschoben werden soll.

Eine Unterbrechungsanforderung wird durch Low-Pegel am Anschluß 19 a eingegeben. Sie wird vom Mikroprozessor nur dann akzeptiert, wenn das Unterbrechungs-Flipflop „INTE" (Interrupt Enable) gesetzt ist. Dies wird durch L-Pegel an Leitung 21 a angezeigt. Das Unterbrechungs-Flipflop kann nur durch „Software" mit dem Befehl „EI" gesetzt werden.

Der Mikroprozessor führt dann noch den Befehl durch, den er gerade abarbeitet, sodann speichert er die nachfolgenden Adressen im Kellerspeicher (Stack). Die Unterbrechung löst also einen ähnlichen Vorgang wie ein Unterprogramm aus. Ist der Befehl abgearbeitet und die nachfolgende Adresse im Kellerspeicher abgespeichert, wird das Signal INTA (Interrupt acknowledge) an Anschluß 23 a abgegeben. Dies ist das Signal für das die Unterbrechung anfordernde Gerät, nunmehr eine RST-Adresse auf den Datenbus zu geben. Bei dieser Adresse steht jetzt das Anwenderprogramm für die Unterbrechung.

In unserem Mikroprozessor-Modul haben wir die Leitung 23 a über einen Widerstand von 1 kΩ mit +12 V verbunden. Diese Verbindung löst im Baustein 8228 zum Zeitpunkt des $\overline{\text{INTA}}$-Signals auf dem Datenbus die Daten FF aus. FF ist der Befehl für die RST-

** Anfragen über Preise und Bezugsmöglichkeiten richten Sie bitte an die nächste Zweigniederlassung der Siemens AG, Abt. VB.

*** Nur bei FUNKSCHAU-Moduln.

Adresse 7. Das Programm wird jetzt bei der Hex-Adresse 38 (Dezimal 56) fortgesetzt. In dieser Adresse muß jetzt das Unterprogramm für die Unterbrechung stehen, bzw. wie in unserem Falle ein Sprungbefehl, der zu der Adresse des Unterbrechungs-Unterprogramms führt.

Sollen in gewissen Teilbereichen des Hauptprogramms aus irgendwelchen Gründen Unterbrechungen nicht akzeptiert werden, so muß vor diesem Programmteil der Befehl „DI" (Disable Interrupts) stehen, nach Beenden dieses Teilbereiches erneut der Befehl „EI", der das INTE-Flipflop setzt.

Zur Bearbeitung von Programm-Unterbrechungen müssen im Hauptprogramm vier Dinge beachtet werden:

1. Der Stapelzeiger (Stack-pointer) muß mit einer Adresse geladen werden, die in unserem RAM-Speicherbereich verfügbar ist. Haben Sie nur die erste Ausbaustufe nach Bild 23 gewählt, ist dies beispielsweise die Adresse ØØFF.
   Haben Sie bisher nur die Ausbaustufe mit einem PROM nach Bild 24 benutzt, müssen Sie Ihre Schaltung nach Bild 22 um ein Speicherpaar SAB 8111 und einen Decoderbaustein SAB 8205 erweitern. Hier käme dann für den Stapelzeiger die Adresse 1ØFF in Frage.
2. Im Programm muß vor der Stelle, an der Sie eine Unterbrechungsanforderung akzeptieren wollen, einmal der Befehl „EI" gestanden haben.
3. Das Unterprogramm für die Programm-Unterbrechung muß auf der Hexa-Dezimal-Adresse ØØ38 stehen. In unserem Falle steht dort ein Sprungbefehl zur Adresse ØØ60.
4. Es ist zu beachten, daß nach einer Unterbrechungs-Anforderung das INTE-Flipflop rückgesetzt ist, d. h. wenn wir in unserem Programm weitere Unterbrechungen zulassen wollen, muß entweder am Ende des Unterprogramms der Befehl „EI" stehen oder im Hauptprogramm dieser Befehl während des Programmzyklus immer wieder durchlaufen werden.
4. In unserem Falle steht der Befehl „EI" als vorletzter Befehl im Unterprogramm vor dem Befehl RETURN.

Unterprogramme

Unterprogramme werden mit dem Befehl „CALL" aufgerufen. Der Befehl arbeitet wie ein Sprungbefehl und verursacht eine Veränderung der Programmsteuerung. Zusätzlich wird die Rücksprungadresse, die für den späteren RETURN-Befehl gebraucht wird, im Kellerspeicher abgelegt.

Unterprogramme werden im allgemeinen immer dann verwendet, wenn

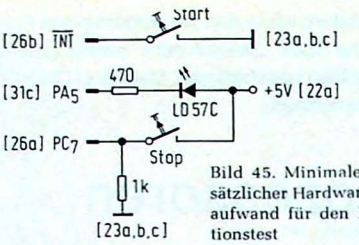


Bild 45. Minimaler zusätzlicher Hardwareaufwand für den Reaktionstest

gleichartige Programmteile mehrmals in einem Programm vorkommen. Neuerdings geht man bei der strukturierten Programmierung dazu über, ein Hauptprogramm aus mehreren gut übersehbaren Unterprogrammen zusammenzusetzen. Die Unterprogramme können dann an verschiedenen Speicherstellen untergebracht sein. Werden im Unterprogramm für die Operationen verschiedener Register gebraucht, so ist es sinnvoll, den Inhalt dieser Register, den Sie beim Verlassen des Hauptprogramms hatten, im Kellerspeicher abzuspeichern und kurz vor dem Rücksprung wieder dort einzulesen. Auf jeden Fall ist es erforderlich, das Programmstatuswort „PSW", das ist der Inhalt des Akkumulators, und die Zustände der verschiedenen Flags im Kellerspeicher abzulegen. Die Ablage geschieht mit dem Befehl „PUSH" und die Rückführung mit dem Befehl „POP". Es ist darauf zu achten, daß die „PUSH"- und „POP"-Befehle spiegelbildlich im Programm stehen, d. h. der zuletzt abgespeicherte Inhalt eines bestimmten Registers der dann als letztes in den Kellerspeicher eingeschrieben wurde, muß als erster wieder zurückgeführt werden.

Reaktionszeit-Test

Der zusätzliche „Hardware"-Aufwand für ein solches Testgerät mit einem Mikrocomputer (*Bild 45*) ist relativ gering. Die Anschluß-Nummern in eckigen Klammern beziehen sich auf die Kopfleiste (*nicht auf die Rückwandverdrahtung*)!! Es werden zwei Tasten, eine Leuchtdiode, zwei Widerstände benötigt. Mit der Taste „Start" wird die Unterbrechungsanforderung INT an den Mikroprozessor gegeben. Im Programm ist eine Verzögerung eingebaut, deren Länge vom Spielstand des Spieles „Seeschlacht" abhängt, also nahezu unbestimmt ist. Wenn die LED LD 57 C aufleuchtet, ist so schnell wie möglich die Taste „Stop" zu drücken. Das Ergebnis steht dann in der Speicherstelle ØØAF, welche über den Anzeige-Modul und die Hilfsschaltung zur Anzeige von Speicherinhalten während des Betriebes angezeigt werden kann. Die gleiche Adresse kann auch bei Benutzung eines PROMs als Programmspeicher gewählt werden, obwohl man nicht in die Speicherzellen einschreiben kann, da der Hexadezimal-7-Segment-Decoder 9368 im Anzeigemodul mit einem Latch ausgerüstet ist und damit die Anzeige speichern kann.

```
0060                    ORG   0060H
0060 F5        REAT:    PUSH  PSW
0061 DBEE               IN    0EEH
0063 5F                 MOV   E,A
0064 C5                 PUSH  B
0065 D5                 PUSH  D
0066 3E89               MVI   A,89H
0068 D3EF               OUT   0EFH
006A 3E20               MVI   A,20H
006C B1                 ORA   C
006D D3EC               OUT   0ECH
006F 78                 MOV   A,B
0070 D3ED               OUT   0EDH
0072 3E00               MVI   A,0
0074 32AF00             STA   00AFH
0077 CDB000             CALL  VERZ0
007A DBEC               IN    0ECH
007C E6DF               ANI   0DFH
007E D3EC               OUT   0ECH
0080 0600               MVI   B,0
0082 0E2A               MVI   C,42D
0084 1600               MVI   D,0
0086 05        RZEI:    DCR   B
0087 C28600             JNZ   RZEI
008A 0D                 DCR   C
008B C28600             JNZ   RZEI
008E 0E2A               MVI   C,42D
0090 14                 INR   D
0091 DBEE               IN    0EEH
0093 E680               ANI   80H
0095 CA8600             JZ    RZEI
0098 7A                 MOV   A,D
0099 27                 DAA
009A 32AF00             STA   00AFH
009D D1                 POP   D
009E C1                 POP   B
009F 3E81               MVI   A,81H
00A1 D3EF               OUT   0EFH
00A3 79                 MOV   A,C
00A4 D3EC               OUT   0ECH
00A6 78                 MOV   A,B
00A7 D3ED               OUT   0EDH
00A9 7B                 MOV   A,E
00AA D3EE               OUT   0EEH
00AC F1                 POP   PSW
00AD FB                 EI
00AE C9                 RET
00B0           VERZ0    EQU   00B0H
0000                    END
```

Bild 46. Unterbrechungsunterprogramm für den Reaktionstest

```
00B0                    ORG   00B0H
00B0 1605      VERZ0:   MVI   D,05
00B2 05        VERZ1:   DCR   B
00B3 C2B200             JNZ   VERZ1
00B6 0D                 DCR   C
00B7 C2B200             JNZ   VERZ1
00BA 15                 DCR   D
00BB C2B200             JNZ   VERZ1
00BE C9                 RET
0000                    END
```

Bild 47. Unterprogramm für die Verzögerung

Der Adreßschalter muß auf Adresse ØØAF stehen und der E/A-Speicher (siehe 3. Teil, Bild 17) auf „Speicher". Dann werden die Daten, die in die Speicherzelle 00AF eingeschrieben werden sollen, angezeigt, egal, ob ein Schreib/Lesespeicher angeschlossen ist oder nicht. Als „Software" sind zusätzlich zum Programm „Seeschlacht" ein Unterbrechungs-Programm (*Bild 46*) und ein Unterprogramm (*Bild 47*) erforderlich. *(Fortsetzung folgt)*

Horst Pelka

Im vorliegenden 15. Teil beschreibt der Autor einige kleinere Schaltungen und Programmänderungen, mit deren Hilfe die Unterprogramme in das bestehende System eingefügt werden.

Praxis mit Mikroprozessoren

15. Teil

Beide Programme sind jetzt für die Verwendung eines batteriegepufferten CMOS-RAMs ausgelegt. Wird ein PROM verwendet, so müssen die Speicherstellen für den Kellerspeicher in dem Adreßbereich eines RAMs liegen, welches zusätzlich in die Speicherplatine eingebaut werden muß. Bei Schaltung nach Bild 22 (4. Teil) waren dies je nach Bestückungsgrad Adresse 10FF, 11FF, 12FF oder 13FF.

Wir sind jetzt im Besitz eines Doppelspieles, einmal „Seeschlacht" und einmal „Reaktionstest". Das Spiel „Seeschlacht" wird durch Drücken der Starttaste des Reaktionstestes für die Zeit des Testes unterbrochen. In der vorher angegebenen Form sind jedoch zwei kleine „Schönheitsfehler" vorhanden.

Der eine Schönheitsfehler ist, daß mit der RESET-Taste der Reaktionstest jeweils unterbrochen werden kann, so daß die erste Spielpartei die zweite Partei behindern kann. Eine gewisse Behinderung der ersten Partei durch den Reaktionstest war ja absichtlich gegeben.

Der zweite Schönheitsfehler ist, daß die Lampe, die den Beginn des Reaktionstestes kennzeichnet, auch ohne Drücken der Starttaste brennen kann; dies hängt vom zufälligen Ablauf des Spieles „Seeschlacht" ab. Sie kann also entweder brennen oder nicht brennen. Bei Beginn des Spieles „Seeschlacht" wird sie immer brennen, ansonsten kennzeichnet das Verlöschen einen Treffer eines während des Spieles aber nicht angezeigten Objektes. Die Schaltung nach *Bild 48* vermeidet beide Nachteile.

Ein Kennzeichen dafür, daß das Spiel „Seeschlacht" gerade gespielt wird, ist, daß eine der an die Klemmen PC 5 oder PC 4 angeschlossenen Lumineszenz-Dioden brennt, oder beide Lampen brennen bei Beginn des Spieles. Wird das Spiel durch einen Reaktionstest (Programm-Unterbrechung) unterbrochen, so schaltet dieses Unterbrechungsprogramm beide Lampen aus. Diese Tatsache kann dazu benutzt werden, die Reset-Taste elektrisch zu verriegeln.

Über ein „Software"-Programm ist dies bei einem Mikroprozessor leider nicht möglich. Beide Klemmen, PC 4 und PC 5, gehen auf den Eingang eines NAND-Gatters, welches nur dann am Ausgang L-Pegel führt, wenn beide Klemmen auf H-Pegel sind. Dies ist der Fall, wenn beide Lampen ausgeschaltet sind. In diesem Falle gibt das zweite NAND-Gatter am Ausgang, der zum Anschluß 22 c der Kopfleiste führt, immer H-Signal ab, ganz egal, welcher Pegel auf den zweiten Eingang dieses NAND-Gatters führt. Führt der Eingang des ersten NAND-Gatters an einem oder beiden Eingängen L-Pegel, so ist sein Ausgang auf H und damit das zweite NAND-Gatter derart aktiviert, daß beim Anlegen von H-Pegel an den zweiten Eingang der nach Klemme 22 c führende Ausgang L-Pegel führt und damit ein RESET auslöst. Erforderlich ist lediglich noch ein Inverter, welcher den L-Pegel bei gedrückter RESET-Taste in den gewünschten H-Pegel umwandelt.

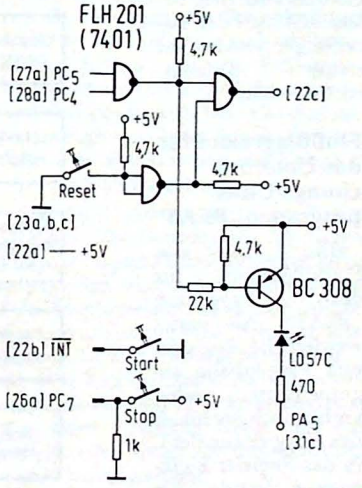


Bild 48. Schaltung zur Reset-Sperre während des Unterprogrammes (oben) und durch Lampenverriegelung erweiterte Schaltung nach Bild 45 (unten)

Die gleiche Logik kann mit einem Zusatzaufwand von einem Transistor und zwei Widerständen dazu benutzt werden, die Leuchtdiode für den Reaktionstest nur dann einschalten zu können, wenn der Reaktionstest in Betrieb ist. In diesem Falle führen die beiden Klemmen PC 4 und PC 5 H-Pegel; am Ausgang des NAND-Gatters ist L-Pegel, welcher über die Widerstände 22 k/4,7 k den Transistor BC 308 leitend steuert. Wird vom Programm her die Leuchtdiode eingeschaltet, führt die Anschluß-Klemme PA 5 L-Pegel; es kommt ein Stromfluß von +5 V über den Transistor, die Lumineszenz-Diode und den Strombegrenzungswiderstand 470 Ω zur nahezu auf Masse liegenden Klemme PA 5 zustande.

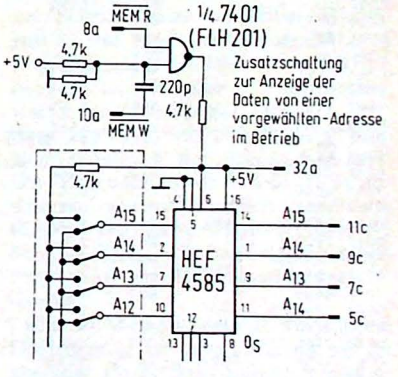


Bild 49. Einbau eines zusätzlichen RC-Gliedes in Schaltung nach Bild 17 (3. Teil) zur Verkürzung des $\overline{\text{MEMW}}$-Impulses

Bauteileliste:

1 Lumineszenz-Diode LD 57 C (Siemens AG)
1 Widerstand 470 Ω
1 Widerstand 1 kΩ
2 einpolige Tasten

Bauteileliste:

1 Baustein 4 NAND-Glieder mit je zwei Eingängen und offenem Kollektor FLH 201 (Siemens AG)
1 Transistor BC 308 (Siemens AG)
1 Lumineszenz-Diode LD 57 C (Siemens AG)
1 Widerstand 470 Ω
1 Widerstand 1 kΩ
4 Widerstände 4,7 kΩ
1 Widerstand 22 k
2 einpolige Tasten
1 einpolige Taste für Reset (war bereits im Bedienteil vorhanden).

Bauteileliste:

1 Styroflex-Kondensator 220 pF, 25 V – B 31310 – A 3221 – H (Siemens AG)
2 Widerstände 4,7 kΩ

Bei einigen Mikroprozessor-Exemplaren kann es im Zusammenspiel mit dem Anzeige-Modul zu Schwierigkeiten des Auslesens von Adressen und Daten beim „Memory write"-Modus während des Betriebes kommen, weil der Steuerimpuls MEMW länger anhält, als gültige Adressen und Daten auf Adreßbus und Datenbus stabil sind. Als Abhilfe wird empfohlen, diesen MEMW-Impuls durch ein RC-Glied zu verkürzen (*Bild 49*).

Änderungen am Programm „Seeschlacht" zur Ausführung des Unterbrechungs-Unterprogramms

Für die Akzeptierung und Durchführung des Unterprogramm-Aufrufes müssen wir jetzt im Hauptprogramm (*Bild 50*) an einigen Speicherstellen eine Änderung vornehmen. Wir benutzen dazu Speicherstellen, die aus didaktischen Gründen Teil 10 und 11 zunächst redundante Befehle enthielten. Da der Befehl auf Adresse ØØØB ein 3-Byte-Befehl ist, füllt er also die Plätze ØØØB, ØØØC und ØØØD. Der nachfolgende Befehl auf Adresse ØØØE ist ein 1-Byte-Befehl, so daß auf der Adresse ØØØF wieder das alte Programm weitergeht. Der Befehl auf Adresse ØØ38 ist ein 3-Byte-Befehl. Er füllt die Plätze ØØ38, ØØ39 und ØØ3A.

Da bei einer Programmunterbrechung bereits die im Programm laufende Adresse im Kellerspeicher abgespeichert wird, wird hier als Sprungbefehl der Befehl „JMP" benutzt (und nicht der Befehl „Call"!!). Damit ist sichergestellt, daß am Ende des Unterbrechungs-Unterprogramms nach dem Befehl RETURN der Stapelzeiger wieder auf seiner Anfangsadresse steht.

```
0000                    ORG     0000
0000 00                 NOP
0001 00                 NOP
0002 00                 NOP
0003 3E81               MVI     A,81H
0005 D3EF               OUT     0EFH
0007 3E00               MVI     A,00
0009 D3EC               OUT     0ECH
000B 31FF00             LXI     SP,00FFH
000E FB                 EI
000F 0600               MVI     B,00
0011 0E00               MVI     C,00
0013 1601               MVI     D,01H
0015 DBEE       ABF1:   IN      0EEH
0017 E602               ANI     02H
0019 C24300             JNZ     SPI2
001C DBEE               IN      0EEH
001E E608               ANI     08H
0020 CA1500             JZ      ABF1
0023 3EE0       SPI1:   MVI     A,0E0H
0025 D3EE               OUT     0EEH
0027 DBEE       STA1:   IN      0EEH
0029 E601               ANI     01H
002B CA2700             JZ      STA1
002E 7A         SCH1:   MOV     A,D
002F 0F                 RRC
0030 57                 MOV     D,A
0031 DBEE       ST01:   IN      0EEH
0033 E602               ANI     02H
0035 C33B00             JMP     FORT
0038 C36000             JMP     0060H
003B CA2E00     FORT:   JZ      SCH1
003E 7A                 MOV     A,D
003F B0                 ORA     B
0040 D3ED               OUT     0EDH
0042 47                 MOV     B,A
0043 3ED0       SPI2:   MVI     A,0D0H
0045 D3EE               OUT     0EEH
0047 DBEE       STA2:   IN      0EEH
0049 E604               ANI     04H
004B CA4700             JZ      STA2
004E 7A         SCH2:   MOV     A,D
004F 0F                 RRC
0050 57                 MOV     D,A
0051 DBEE       ST02:   IN      0EEH
0053 E608               ANI     08H
0055 CA4E00             JZ      SCH2
0058 7A                 MOV     A,D
0059 B1                 ORA     C
005A D3EC               OUT     0ECH
005C 4F                 MOV     C,A
005D C32300             JMP     SPI1
0060                    END
```

◄ Bild 50. Änderung des Programmes nach Tabelle 1 (Folge 5) zur Akzeptierung von Unterbrechungen mit Sprungbefehl nach Adresse ØØ6Ø auf Adresse ØØ38

Flußdiagramm für das Unterbrechungs-Unterprogramm „REAT"

Bild 51 zeigt das Flußdiagramm für das Programm „REAT". Als erstes werden die Register- und Flag-Inhalte abgespeichert, wobei vorher noch der Inhalt des Ausgabekanales C in das Register E eingelesen wurde. Als nächstes wird der Eingabe/Ausgabe-Baustein so umprogrammiert, daß Kanal A und B weiterhin als Ausgabe, Kanal C aber in der vollen Breite von 8 bit als Eingabe arbeitet. Da bei der Umprogrammierung des Bausteines die Zwischenspeicher für die Ausgänge gelöscht werden, wird als nächstes der Kanal A und Kanal B erneut ausgegeben, um den letzten Stand vor dem Unterbrechungsprogramm wieder herzustellen. Durch eine ODERierung mit dem Bitmuster 20 wird dabei sichergestellt, daß die Lampe für den Reaktionstest ausgeschaltet ist.

Sodann wird die Ausgabespeicherstelle auf Null gesetzt. Das Unterprogramm „VERZO" – ein Unterprogramm für eine Verzögerung – wird aufgerufen. Nach Rückkehr wird die Startlampe eingeschaltet, die Register B und D auf Null sowie das Register C auf 42 (dezimal) gesetzt.

(Fortsetzung folgt)

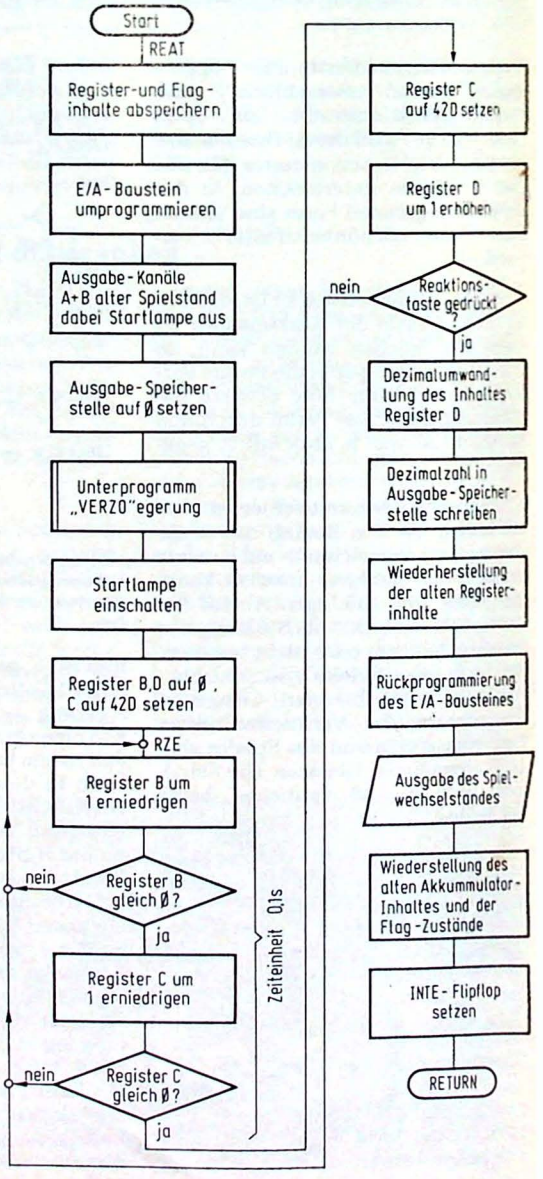


Bild 51. ► Flußdiagramm für das Unterbrechungsunterprogramm „REAT" (Reaktionszeittest)

In 16. Teil dieser Reihe schließt der Verfasser die Beschreibung der vorhergehenden Unterprogramme ab. Anschließend folgt die Beschreibung des Unterprogramms Reaktionszeittest.

Horst Pelka

Praxis mit Mikroprozessoren

16. Teil

Eine Schleife, die einmal das Register B jeweils um eins erniedrigt, dann abfragt, ob der Inhalt des Registers B gleich Null ist, wird 256mal durchlaufen, da man beim ersten Durchlauf vom Registerinhalt Null, der vorher gesetzt worden war, durch Erniedrigung um eins auf den Register-Inhalt 255 gekommen war. Als nächstes führt die Verzweigung, wenn der Register-Inhalt des Registers B gleich Null gewesen ist, den Befehl, das Register C um eins zu erniedrigen, aus. Der Inhalt wird also von 42 auf 41 kommen. Dies geschieht so lange in einer etwas größeren Schleife, wobei beim Durchlaufen der großen Schleife jedesmal 256 kleine Schleifen enthalten sind, bis der Register-Inhalt C gleich Null ist. Durch diese beiden Schleifen und den vom Mikroprozessor gegebenen Taktintervall wird eine Zeiteinheit von 0,1 s geformt. Ist diese Zeiteinheit einmal durchlaufen, wird das Register D um eins erhöht. Im Register D stehen also immer Zeiten mit der Zeiteinheit 0,1 s.

Sodann wird die Reaktionstaste abgefragt. Ist sie nicht gedrückt, durchläuft das Programm wieder diese Zeitschleife 0,1 s, um erneut auf Abfrage zu gehen. Erst wenn die Reaktionstaste gedrückt war, kann das Programm zur Dezimalumwandlung des Inhaltes des Registers D fortfahren. Die Dezimalzahl wird in die Ausgabespeicherstelle geschrieben.

Um immer wieder die korrekte Zeiteinheit zu bekommen, wird nach dem Vergleich, „Register C gleich Null?" das Register C sofort wieder auf 42 (dezimal) gesetzt.

Nach der Ausgabe der Dezimalzahl auf die Speicherstelle, in unserem Falle mit der Adresse ØØAF werden die alten Register- und Flagzustände wiederhergestellt. Der Eingabe/Ausgabe-Baustein wird auf seine ursprüngliche Konfiguration rückprogrammiert und der vor der Unterbrechung gewesene Spielstand und die Anzeige, welcher Spieler an der Reihe ist, wird ausgegeben. Sodann wird der Inhalt des Akkumulators und der Flags vor der Unterbrechung wiederhergestellt und der Interrupt-Enable-Flipflop gesetzt, so daß eine erneute Unterbrechung möglich ist. Als letztes wird das Unterprogramm mit dem Befehl RETURN abgeschlossen.

Flußdiagramm für das Unterprogramm „VERZO"

Der Inhalt des Registers D wird auf Ø5 gesetzt (*Bild 52*), als nächstes wird das Register B dekrementiert und in einer Verzweigung abgefragt, ob der Register-Inhalt Null ist. Beim Register-Inhalt Null wird das Programm mit der

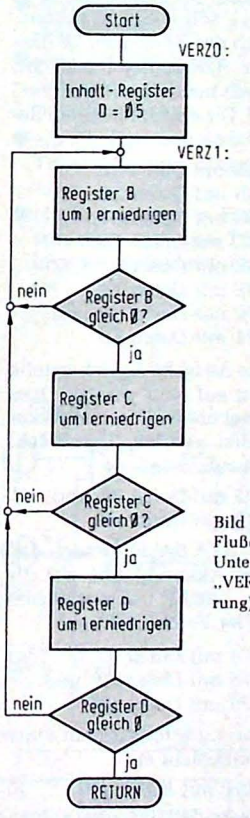


Bild 52. Flußdiagramm für das Unterprogramm „VERZO" (Verzögerung)

Dekrementierung des Registers C fortgesetzt und wieder abgefragt, ob das Register C den Inhalt Null hat. Dabei bildet sich wieder eine kleine und eine große Schleife, wobei jeweils bei der Dekrementierung des Registers C um 1 wieder 256 Durchläufe zum Dekrementieren des Registers B absolviert werden müssen.

Erst wenn das Register C gleich Null ist, wird das Programm mit dem Dekrementieren des Registers D fortgeführt. In diesem Falle bildet sich eine noch größere Schleife, wobei das Register C 256mal dekrementiert wird, wenn eine Dekrementierung des Registers D erfolgt. Da das Register D vorher auf Ø5 gesetzt ist, wird diese große Schleife nur fünfmal durchlaufen. Ist der Inhalt des Registers D gleich Null, geht die Verzweigung auf den letzten Befehl, der ein Rücksprungbefehl RET ist. Dadurch, daß von dem vorhergehenden Hauptprogramm die Register-Inhalte B und C einen unbestimmten Inhalt haben, ist die Länge dieser Verzögerungszeit nicht direkt vorhersehbar. In einer späteren Folge wird noch ein Programm beschrieben, bei dem die Verzögerung völlig unabhängig mit einem Unterprogramm für einen Zufallsgenerator erzeugt wird.

Unterbrechungs-Unterprogramm, Reaktionszeittest (REAT)

Der Sprungbefehl in Adresse ØØ38 bis ØØØA führt auf die Adresse ØØ6Ø. Der erste Befehl „PUSH" PSW speichert den Inhalt des Akkumulators und die Zustände der Flags im Kellerspeicher ab. Er steht in

Adresse ØØ6Ø mit den Daten F5.

Wenn der Stapelzeiger auf die Adresse ØØFF festgelegt wurde, erscheint als nächstes die Adresse ØØFF – 1. Das ist

Adresse ØØFC [1]) mit Daten XX.

Dies sind die Daten der fünf Bedingungs-Bits (Flags). In der nächsten Kellerspeicheradresse

Adresse ØØFB [1]) mit Daten XX

wird der Inhalt des Akkumulators im Kellerspeicher abgespeichert. Da zunächst nicht voraussehbar war, an welcher Stelle das Programm durch den Interrupt unterbrochen wurde, kann hier auch der Inhalt der fünf Flags und des Akkumulators nicht festgelegt werden.

Als nächstes soll der Spielstand, d. h. welcher Spieler gerade an der Reihe war, festgehalten werden. Da dieser Spielstand bisher nicht in einem Register abgelegt war, sondern die Ausgänge des Kanales C einfach umgeschaltet wurden, stellen wir den Zustand der Ausgänge des Kanals C fest.

[1]) Für Kellerspeicher mit Stapelzeigeranfangsadresse 1ØFF gelten sinngemäß die um 1000 erhöhten Adressen.

Dies können wir mit einem Einlesebefehl in ØEEH feststellen. Der zugehörige Befehl steht auf

Adresse ØØ61 mit den Daten DB und
Adresse ØØ62 mit den Daten EE.

Bei der nun folgenden Eingabe werden auf

Adresse EEEE die Daten DØ oder EØ

erscheinen.

Dieser Spielstand wird jetzt mit dem Befehl MOV E,A auf

Adresse ØØ63 mit Daten 5F

in das Register E übertragen.
Als nächstes werden die Inhalte der Register B und C in dem Kellerspeicher abgelegt. Der Befehl „PUSH B" steht auf

Adresse ØØ64 mit Daten C5.

Der Inhalt des Registers B wird im Kellerspeicher mit der

Adresse ØØFA [1]) und Daten XX

abgelegt. Der Inhalt des Registers C kommt in den Kellerspeicher mit

Adresse ØØF9 [1]) mit Daten XX.

Auch hier wieder hängt der Inhalt dieser beiden Registerpaare von der Stelle ab, bei der das Programm unterbrochen wurde.

Dann werden die Inhalte der Register D und E abgelegt mit dem Befehl „PUSH D" auf

Adresse ØØ65 mit Daten D5.

Der Inhalt des Registers D wird im Kellerspeicher mit der

Adresse ØØF8 [1]) mit Daten XX,

der Inhalt des Registers B in den Kellerspeicher mit der

Adresse F7 [1]) und Daten XX

abgelegt. Auch hier ist wieder nicht vorhersehbar, welche Daten gerade die Register bei Programmunterbrechung hatten.

Da wir für den Reaktionstest einen neuen Eingang brauchen, bisher aber bereits beim Kanal E die vier unteren Bits mit Eingängen belegt sind, programmieren wir den Eingabe/Ausgabe-Baustein derart um, daß alle Klemmen des Kanales C als Eingänge fungieren. Dies geschieht mit dem Steuerwort 89, welches zunächst in den Akkumulator mit dem Befehl „MVI A,89 H" auf

Adresse ØØ66 mit Daten 3E und
Adresse ØØ67 mit Daten 89

geladen wird. Der nächste Befehl ist ein Ausgabebefehl, der den Inhalt des Akkumulators in das Steuerwortregister des E/A-Bausteines einschreibt. Er lautet „OUT ØEFH" und steht auf

Adresse ØØ68 mit Daten D3 und
Adresse ØØ69 mit Daten EF.

Der Mikroprozessor führt jetzt diese Ausgabe durch. Es sind daher auf

Adresse EFEF die Daten 89

zu sehen. Durch das Umprogrammieren des E/A-Bausteines sind automatisch sämtliche Ausgänge auf L geschaltet worden. Um die Spieler des Spieles „Seeschlacht" nicht zu irritieren, wird die Anzeige des letzten Spielstandes wiederhergestellt, indem der Inhalt des Registers C auf Kanal A und der Inhalt des Registers B auf Kanal B ausgegeben wird. Gleichzeitig soll sichergestellt werden, daß die Lampe, welche den Beginn des Reaktionszeittestes anzeigt, jetzt nicht brennt. Ihr Anschluß, d. h. die Klemme P_{A5}, soll H-Signal führen. Dies wird durch das Bit-Muster 2Ø (Hexadezimal) im Akkumulator erreicht, welches mit dem Inhalt des Registers C ODERiert wird. Die entsprechenden Befehle stehen auf

Adresse ØØ6A mit Daten 3E,
Adresse ØØ6B mit Daten 2Ø,
Adresse ØØ6C mit Daten B1,
Adresse ØØ6D mit Daten D3,
Adresse ØØ6E mit Daten EC,
Adresse ØØ6F mit Daten 78,
Adresse ØØ70 mit Daten D3 und
Adresse ØØ71 mit Daten ED.

Jetzt wird die Ausgabe-Speicherstelle ØØAF zunächst auf Null gesetzt. Dazu muß erst einmal die Null in den Akkumulator geladen werden. Der Befehl MVI A,Ø steht auf

Adresse ØØ72 mit Daten 3E und
Adresse ØØ73 mit Daten ØØ.

Der Befehl „STA ØØAF" besagt, daß der Inhalt des Akkumulators auf die Speicherstelle ØØAF abgespeichert werden soll. Der Befehl steht auf

Adresse ØØ74 mit Daten 32,
Adresse ØØ75 mit Daten AF und
Adresse ØØ76 mit Daten ØØ.

Seine Ausführung erfolgt durch einen Speicherschreibbefehl auf

Adresse ØØAF mit Daten ØØ.

Das nun folgende Verzögerungsprogramm „VERZO" soll den Start des Reaktionstests auf einen unbestimmten Zeitpunkt bringen. Dazu werden die zufälligen Inhalte der Register B und C herangezogen, die je nach dem Spielstand des Spieles im Hauptprogramm unterschiedliche Inhalte haben können. Dieses Unterprogramm steht auf Adresse ØØBØ und wird mit dem Befehl „CALL VERZO" aufgerufen. Der zugehörige Befehl steht in

Adresse ØØ77 mit Daten CD
Adresse ØØ78 mit Daten B0 und
Adresse ØØ79 mit Daten ØØ.

Durch den Programmaufruf wird die nachfolgende Adresse, in diesem Falle ØØ7A, im Kellerspeicher abgespeichert. Dabei werden auf

Adresse ØØF6 die Daten ØØ und
Adresse ØØF5 die Daten 7A

abgespeichert.

Jetzt wird zuerst das Unterprogramm abgearbeitet. Die nächste Adresse ist daher

Adresse ØØBØ mit Daten 16 und
Adresse ØØB1 mit Daten Ø5.

Es handelt sich um den Befehl „MVI D,Ø5" (Lade Register D mit der Sedezimalzahl Ø5). Die Adressen ØØB2 bis ØØBB beinhalten eine Zeitschleife zur unbestimmten Verzögerung des Einschaltens der Lampe und damit den Start zum Reaktionstest. Dabei haben die Register B und Register C einen unbestimmten Inhalt, der sich aus dem Stand des vorhergehenden Spieles ergibt. Beide Register werden in Schleifen heruntergezählt. Außerdem wird das mit Ø5 geladene Register D heruntergezählt, erst dann ist das Unterprogramm beendet, und es erfolgt eine Rückkehr zum durch dieses Unterprogramm unterbrochenen Programm. Man sieht daraus, daß beispielsweise auch ein Unterprogramm wieder durch ein weiteres Unterprogramm unterbrochen werden kann; dies kann beliebig ineinandergeschachtelt sein.

Die erste Schleife umfaßt die Adressen ØØB2 bis ØØB5. Der erste Befehl „DCR B" auf

Adresse ØØB2 mit Daten Ø5

dekrementiert bei jedem Durchlauf das Register B. Der nachfolgende Befehl „JNZ VERZ1" auf

Adresse ØØB3 mit Daten C2
Adresse ØØB4 mit Daten B2
Adresse ØØB5 mit Daten ØØ

läßt das Programm so lange zur Adresse ØØB2 zurückspringen, bis der Inhalt des Registers B Null geworden ist. Erst dann wird der nächste Befehl „DCR C" erreicht, der das Register C herunterzählt. Er ist auf

Adresse ØØB6 mit Daten ØD

untergebracht. Der nachfolgende Befehl „JNZ VERZ1" auf

Adresse ØØB7 mit Daten C2
Adresse ØØB8 mit Daten B2 und
Adresse ØØB9 mit Daten ØØ

läßt das Programm immer wieder nach Adresse ØØB2 zurückspringen, bis das Register C den Inhalt Null hat. Jetzt beginnt das Herunterzählen des Registers D mit dem Befehl „DCR D" auf

Adresse ØØBA mit Daten 15.

(Fortsetzung folgt

Horst Pelka

In der vorliegenden 17. Folge schließt der Verfasser die Beschreibung des Unterprogramms Reaktionszeittest ab.

Praxis mit Mikroprozessoren

17. Teil

Der nachfolgende Befehl schließt wieder die Schleife so lange, bis das Register D den Wert Null erreicht hat. Er steht auf

Adresse ØØBB mit Daten C2
Adresse ØØBC mit Daten B2 und
Adresse ØØBD mit Daten ØØ.

Hat das Register D den Wert Null erreicht, kann das Programm zur

Adresse ØØBE mit Daten C9

gelangen. Der Befehl „RET" ist ein Rückkehrbefehl zum vorhergehenden Programm. Es wird jetzt zunächst die Rückkehradresse aus dem Kellerspeicher geholt, d. h. aus

Adresse ØØF5 die Daten 7A und
Adresse ØØF6 die Daten ØØ.

Jetzt wird das Programm auf

Adresse ØØ7A mit Daten DB und
Adresse ØØ7B mit Daten EC

fortgesetzt. Der letzte Eingabe-Befehl „lädt den Akkumulator mit den Daten des Kanales A". Die nachfolgende UNDierung mit dem Bit-Muster DF und anschließende Ausgabe bringt die Klemme PA5 des Kanales A auf L-Pegel und schaltet damit die Lampe als Zeichen des Beginnes des Reaktionstestes ein. Die Befehle stehen auf den

Adressen ØØ7C mit Daten 6E
Adresse ØØ7D mit Daten DF
Adresse ØØ7E mit Daten D3
Adresse ØØ7F mit Daten EC.

Es erscheint nach Adresse ØØ7B die Eingabe mit

Adresse ECEC und Daten XX,

die vom augenblicklichen Stand des Kanales A abhängen, dann erfolgt die UNDierung, nach dem Befehl auf Adresse ØØ7F erfolgt die Ausgabe mit

Adresse ECEC und Daten XX,

die vom Spielstand des Kanales A abhängen. Als Vorbereitung für die jetzt folgenden Zeitschleifen werden die Inhalte der Register B, C und D auf definierte Werte gesetzt. Und zwar das Register B auf Null, das Register C auf dezimal 42 = hexadezimal 2A und das Register D auf Null. Diese Befehle stehen auf

Adresse ØØ8Ø mit Daten Ø6
Adresse ØØ81 mit Daten ØØ
Adresse ØØ82 mit Daten ØE
Adresse ØØ83 mit Daten 2A
Adresse ØØ84 mit Daten 16 und
Adresse 0085 mit Daten ØØ.

Der nächste Befehl dekrementiert das Register B. Da das Register B vorher auf Null gesetzt wird, ist der Inhalt des Registers B nach einmaligem Dekrementieren FF, welches dezimal ausgedrückt 255 ist. Dieser Befehl steht auf

Adresse ØØ86 mit Daten Ø5.

Der nachfolgende Sprungbefehl „JNZ RZEI" läßt so lange das Programm auf Adresse ØØ86 zurückgehen, bis das Register B wieder einen Nulldurchgang zeigt. Wollen wir dieses mit unserem Mikroprozessor in Einzelschritten verfolgen, müßten wir also 256mal die Schleife durchfahren. Es ist daher sinnvoll, zur Beobachtung das Register B nicht mit Null, sondern mit Ø2 zu laden, ferner das Register C nicht mit 2A, sondern mit Ø1, dann kann man diese Schleifen leichter beobachten. Eine andere Hardware-Möglichkeit wird in einer der nächsten Folgen besprochen.

Vergessen Sie aber bitte nicht, wenn Sie das Programm endgültig benutzen wollen, die alte Programmierung wiederherzustellen. Der bedingte Sprungbefehl steht auf

Adresse ØØ87 mit Daten C2
Adresse ØØ88 mit Daten 86 und
Adresse ØØ89 mit Daten ØØ.

Hat das Register B den Inhalt Null erreicht, geht das Programm auf

Adresse ØØ8A mit Daten ØD

weiter. Es handelt sich um den Befehl „DCR C", das Register C wird bei jedem Durchlauf dekrementiert. Der nachfolgende Sprungbefehl läßt das Programm immer wieder zur Adresse ØØ86 zurückgehen, bis auch das Register C den Wert Null erreicht hat, in diesem Falle läuft das Register C aber nicht 256mal durch, sondern nur 42mal. Der dazugehörige Sprungbefehl steht auf

Adresse ØØ8B mit Daten C2
Adresse ØØ8C mit Daten 86 und
Adresse ØØ8D mit Daten ØØ.

Der nachfolgende Befehl schafft wieder die Anfangsbedingung für das Register C und setzt dieses auf dezimal 42 = hexadezimal 2A. Er steht auf

Adresse ØØ8E mit Daten ØE
Adresse ØØ8F mit Daten 2A.

Jetzt wird das Register D vom Inhalt Null hochgezählt. Wenn Sie nach Bild 23 durch Verbinden der Klemmen 11 und 12 a einen Wait-Zyklus eingeschoben haben, ist jeder Durchlauf über

Adresse ØØ9Ø mit Daten 14

in der Schleife 100 ms lang. Es erfolgt also alle 100 ms eine Abfrage, ob die Reaktionstaste auf Klemme PC 7 gedrückt ist. Ist sie nicht gedrückt, erfolgt nach Adresse ØØ97 ein Rücksprung auf Adresse ØØ86, d. h. die Zeitschleife 100 ms wird erneut durchfahren. Dabei zählt das Register D bei jedem Durchgang um 1 weiter, so daß die Anzahl der Schleifen ein Maß für eine Reaktionszeit bis zum Drücken der Taste ist. Die Abfrage, ob die Taste gedrückt ist, geschieht mit den Befehlen „IN ØEEH" und „ANI 8ØH". Sie stehen auf

Adresse ØØ91 mit Daten DB
Adresse ØØ92 mit Daten EE.

Die Eingaben auf

Adresse EEEE mit Daten XX,

je nach dem, ob die Taste gedrückt ist oder nicht und die Selektion der Klemme PC 7 auf

Adresse ØØ93 mit Daten E6 und
Adresse ØØ94 mit Daten 8Ø.

War die Taste gedrückt, ist nach dem UNDieren mit 8ØH der Inhalt des Akkumulators nicht mehr Null, der Sprungbefehl „JZ RZEI" auf

Adresse ØØ95 mit Daten CA
Adresse ØØ96 mit Daten 86 und
Adresse ØØ97 mit Daten ØØ

wird nicht befolgt. Mit dem Befehl „MOV A,D" wird der Inhalt des Registers D zur nachfolgenden Dezimalkorrektur in den Akkumulator geladen. Der Befehl steht auf

Adresse ØØ98 mit Daten 7A.

Die Dezimalkorrektur erfolgt mit dem Befehl „DAA" auf

Adresse ØØ99 mit Daten 27.

Die zu einer Dezimalzahl korrigierte Zeiteinheit (100 ms) steht im Akkumu-

lator und wird mit dem nachfolgenden Befehl „STA ØØAF“ auf

Adresse ØØ9A mit Daten 32
Adresse ØØ9B mit Daten AF
adresse ØØ9C mit Daten ØØ

auf die Speicherstelle ØØAF übertragen. Damit ist der eigentliche Test beendet. Es folgen aber jetzt noch einige Befehle, um nach diesem Unterbrechungs-Unterprogramm den ursprünglichen Zustand des Mikroprozessors wiederherzustellen, ehe wir in das Hauptprogramm zurückgehen. Mit dem Befehl „POP D“ auf

Adresse ØØ9 mit Daten D1

wird der im Kellerspeicher abgespeicherte Inhalt der Register D und E in diese wieder zurückübertragen.
Es wird die

Adresse ØØF7 [1]) *mit Daten XX* und
Adresse ØØF8 [1]) *mit Daten XX*

angezeigt.
Der nächste Befehl „POP B“ auf

Adresse ØØ9E mit Daten C1

bringt den Inhalt der Register B und C in diese zurück.

Adresse ØØF9 [1]) *zeigt Daten XX* und
Adresse ØØFA [1]) *Daten XX.*

Jetzt erfolgt die Rückprogrammierung des Eingabe/Ausgabe-Bausteines 8255, Kanal A und B sowie die höherwertigen Bits des Kanals C als Ausgabe, die niederwertigen Bits des Kanals C als Eingabe. Das zugehörige Steuerwort 81 wird mit dem „MVI A, 81H“ in den Akkumulator geladen. Er steht in

Adresse ØØ9F mit Daten 3E und
Adresse ØØ10 mit Daten 81.

Der nachfolgende Ausgabebefehl „OUT ØEFH“ auf

Adresse ØØA1 mit Daten D3 und
Adresse ØØA2 mit Daten EF

bringt mit

Adresse EFEF und *Daten 81*

das Steuerwort in das Steuerwortregister des Eingabe/Ausgabe-Bausteines. Die Spielstände waren in den Registern B,C und E gespeichert. Wir geben also mit den nachfolgenden Befehlen den Inhalt der Register C auf den Kanal A, den Inhalt des Registers B auf den Kanal B und den Inhalt des Registers E auf den Kanal C aus. Dies wird mit nachfolgenden Befehlen auf

Adresse ØØA3 mit Daten 79,
Adresse ØØA4 mit Daten D3,
Adresse ØØA5 mit Daten EC,
Adresse ECEC mit Daten XX,
Adresse ØØA6 mit Daten 78,
Adresse ØØA7 mit Daten B3,
Adresse ØØA8 mit Daten ED,
Adresse EDED mit Daten XX,
Adresse ØØA9 mit Daten 7B,
Adresse ØØAA mit Daten D3,
Adresse ØØAB mit Daten EE und
Adresse EEEE mit Daten XX

gemacht. Nun muß noch der ursprüngliche Inhalt des Akkumulators zum Zeitpunkt der Programmunterbrechung sowie der Zustand der Flags bei Programmunterbrechung aus dem Kellerspeicher rückübertragen werden. Dies geschieht mit dem Befehl „POP PSW“ auf

Adresse ØØAC mit Daten F1

und Rückführung der Inhalte aus dem Kellerspeicher mit

Adresse ØØFB [1]) *mit Daten XX* sowie
Adresse ØØFC [1]) *MIT Daten XX.*

Da bei Betätigung der Unterbrechungsanforderung das Interrupt-Enable-Flipflop rückgesetzt wird, so daß während der Unterbrechung kein erneuter Interrupt stattfinden kann, muß jetzt noch das Interrupt-Flipflop wieder auf „Enable“ gesetzt werden, so daß eine erneute Unterbrechung möglich ist. Dies tut der Befehl „EI“ auf

Adresse ØØAD mit Daten FB.

Nunmehr kann in das Hauptprogramm zurückgesprungen werden mit dem Befehl „RET“ auf

Adresse 00AE mit Daten C9.

Das Programm setzt sich jetzt an der Stelle fort, wo es durch die Unterbrechungsanforderung unterbrochen wurde. Aus dem Kellerspeicher wird die Fortsetzungsadresse in den Befehlszähler geladen. Da der Stapelzeiger im Falle eines CMOS-RAMs mit der Adresse ØØFF geladen war, kommen jetzt als nächstes die

Adresse ØØFD [1]) *mit Daten XX*
Adresse ØØFE [1]) *mit Daten ØØ*

über den Datenbus in den Mikroprozessor.

Bitte beachten Sie, wenn Sie kein CMOS-RAM als Programmspeicher benutzt haben, daß für den Stapelspeicher eine Adresse eines RAMs angegeben ist, welches in dem Speicher-Modul auf dieser Adresse wirklich vorhanden ist. Ansonsten wird nämlich unser Mikroprozessor aus dem ersten Unterprogramm schon nicht mehr in das Unterbrechungs-Unterprogramm zurückkehren können.

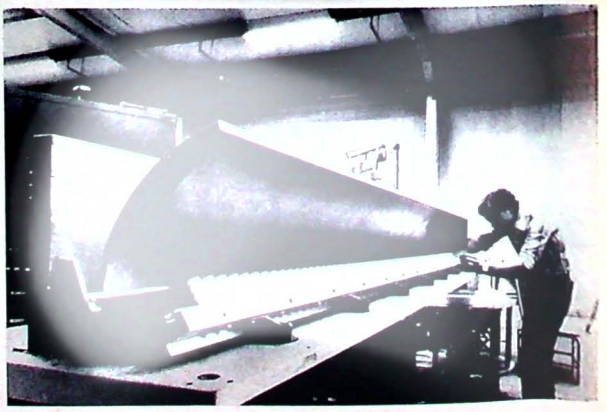
(Fortsetzung folgt in Heft 7)

[1]) Für Kellerspeicher mit Stapelzeigeranfangsadresse 1Ø FF gelten sinngemäß die um 1ØØØ erhöhten Adressen.

Radarkette Jade

Der Schiffsverkehr vor den Küsten, Häfen und deren Zufahrten wird durch ein dichtes Netz von landgestützten Radarstationen überwacht. AEG-Telefunken erhielt den Auftrag zur Lieferung einer Radarkette, die vier Radaranlagen umfaßt und die Jade von dem Gebiet nördlich der Insel Wangerooge bis zum Ölhafen Wilhelmshaven sichern soll.

Um die bei Wasserbeobachtungen sehr starken Seegangs-Störechos zu unterdrücken, wurde ein spezieller Radartyp entwickelt, der diese Aufgabe mit Frequenz- und Polarisations-Diversity-Abtastung löst. Das *Bild* zeigt einen Blick in die 7 m lange lineare Arrayantenne (hier im Prüffeld), bei der die Polarisation elektronisch gesteuert wird. Die Strahlerzeile besteht aus 220 zylinderförmigen Einzelstrahlern und hat bei einer Bündelung von 0,35° eine Reichweite von 24 km. Die Radarsignale werden über eine TV-Richtfunkstrecke in die Kontrollzentrale übertragen.



Diese 7 m lange lineare Arrayantenne für die Radarkette Jade wird auf dem Leuchtturm Wangerooge installiert werden

Horst Pelka

Wenn ein neues Programm geschrieben wird oder ein vorhandenes Programm neu eingegeben wird, kann es natürlich immer wieder zu Fehlern kommen, seien es logische Fehler, oder auch nur Fehler bei der Eingabe der Daten. Bei der Fehlersuche wird man daher so vorgehen, daß man das Programm in Einzelschritten verfolgt.

Praxis mit Mikroprozessoren

18. Teil

„Hardware"-Hilfen bei der Fehlersuche

Man betätigt hierzu den Schalter „Einzelschritte" (single step), vergleicht die bei den Adressen erscheinenden Daten mit dem Programm und schaltet mit der Taste „Step" um einen Schritt weiter. Hat man den Fehler gefunden und das Programm zunächst nur in das CMOS-RAM eingeschrieben, kann man mit direktem Speicherzugriff diese eine Stelle verbessern. Liegt das Programm in einem PROM vor, so muß natürlich das PROM gelöscht und neu programmiert werden.

Manchem ist es lästig, für jeden Schritt die Taste „Step" drücken zu müssen, sondern er möchte beispielsweise im Sekunden- oder 2-Sekunden-Takt einen automatischen Durchlauf des Programms haben. Dies ist mit einem zusätzlichen Schalter nach *Bild 53* möglich, der im eingeschalteten Zustand die IS-Anschlüsse 9 und 13 miteinander verbindet. Wem es zu schnell geht, kann sich dort mit einem zweipoligen Schalter durch Zuschalten einer Kapazität parallel zum 68-µF-Kondensator die Zeitkonstante nach Wunsch einstellen. Beim Ausschalten dieses automatischen Taktes ist zu beachten, daß das Monoflop noch jeweils um einen Schritt weitergeht, da der Auftastimpuls, welcher zur Weiterschaltung benutzt wird, bereits den Kippvorgang ausgelöst hat. Man muß also den Schalter um einen Schritt früher ausschalten, wenn man wieder auf Einzelschritte übergehen will.

Das Verfolgen eines Programms in Einzelschritten ist häufig sehr zeitraubend, wenn im Programm (siehe Bild 47) Zeitschleifen eingebaut sind. Bei einer Taktzeit von 0,5 µs werden natürlich solche Zeitschleifen in einer relativ kurzen Zeit abgearbeitet. Im Sekundentakt würde aber eine solche Zeitschleife von 5x256x256 Schritten, die normalerweise einige Sekunden dauert, ca. 90 Stunden in Anspruch nehmen. Hier sieht man deutlich die Grenzen des Einzelschrittverfahrens. Es ist hier besser, eine andere Methode anzuwenden.

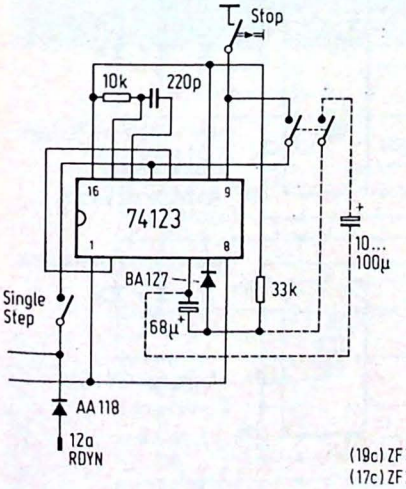


Bild 53. Auszug aus Schaltung nach Bild 3. Hinzugekommen ist der zweipolige Schalter, der mit gestrichelten Leitungen angeschlossene Elektrolytkondensator und die untere Diode AA 118

Setzen von Haltepunkten mittels „Hardware"

Sowohl mit „Software" als auch mit „Hardware" ist es möglich, das Programm an einer bestimmten Stelle zum Anhalten zu bringen. Solche Stopstellen nennt man im englischen Sprachgebrauch „Break points". Als Kriterium für die Unterbrechung kann entweder die Adresse oder auch bestimmte Daten herangezogen werden. In Teil 3, Bild 17, war eine Zusatzschaltung zur Anzeige der Daten einer vorgewählten Adresse im Betrieb gezeigt. Es erscheint immer dann am Anschluß ZF 2 (17 c) ein negativer Impuls, wenn die auf dem Datenbus befindliche Adresse mit der am Datenschalter eingestellten Adresse übereinstimmt. Dieser Impuls setzt ein Flipflop (*Bild 54*), dessen Ausgang über eine Diode AA 118 und den geschlossenen Schalter BP über die RDYN-Leitung den Mikroprozessor anhält. Mit der Taste „L" kann das Flipflop wieder rückgesetzt werden.

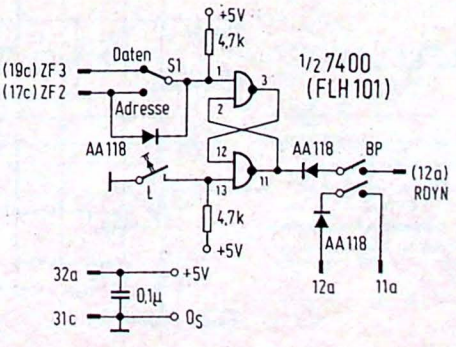


Bild 54. ▶ Zusatzschaltung zum Anhalten des Mikroprozessors bei einer vorgewählten Adresse oder bei vorgewählten Daten

Häufig ist es günstiger, das Programm nicht bei einer bestimmmten Adresse, sondern bei bestimmten Daten-Bytes anzuhalten. Daher ist ein Umschalter zwischen Haltepunkten bei Adressen und Haltepunkten bei Daten vorgesehen.

Die kleine Zusatzschaltung nach *Bild 55* kann irgendwo auf einer Platine mit untergebracht werden. Durch die busartige Verdrahtung über die Rückwandplatine sind die erforderlichen Leitun-

Bauteileliste für Bild 53

1 einpoliger oder doppelpoliger Schalter
1 Elektrolyt-Kondensator (Tantal-Perle)
10...100 µF (je nach gewünschter Taktzeit)
1 Diode AA 118 (Siemens AG)

für Bild 54

1 Baustein 4 NAND-Glieder mit zwei Eingängen FLH 101 (Siemens AG)
3 Germanium-Dioden AA 118 (Siemens AG)
1 Keramik-Kondensator 0,1 µF 63 V = B37449 – A6104 – S2
2 Widerstände 4,7 kΩ
1 zweipoliger Schalter
1 einpoliger Umschalter
1 einpolige Taste

für Bild 55

2 4-Bit-Komparatoren HEF 4585 (Valvo, NSC)
1 Keramik-Kondensator 0,1 µF 63 V = B37449 – A6104 – S2
8 einpolige Umschalter (nur bei Bedarf, wenn Schaltung nicht auf Modul für direkten Speicherzugriff untergebracht wird)

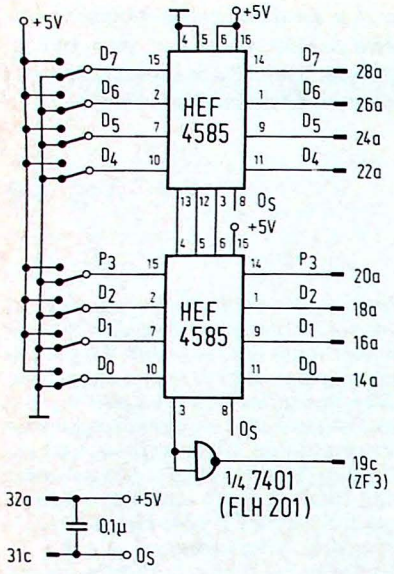


Bild 55. Komparator zum Vergleich der an den Datenschaltern eingestellten und über den Datenbus laufenden Daten. Bei Koinzidenz liegt am Anschluß ZF 3 (19c) L-Signal

gen an jedem Modul verfügbar. Es soll noch darauf hingewiesen werden, daß die Leitungen ZF 2 und ZF 3 mit Hand verdrahtet werden müssen.

Zusätzlich zur Schaltung nach Bild 17 muß noch ein Komparator für die Daten eingebaut werden. Dazu sind lediglich zwei Komparatoren HEF 4585 erforderlich; der Inverter unten im Bild ist beim Baustein 7401 nach Bild 17 noch übrig.

Da die Sample-Schaltung für die Anzeige (siehe Bild 3) vom Anschluß 17c ihr Signal bekommt, muß über eine Diode AA 118 der Impuls bei Gleichheit der an den Datenschaltern eingestellten und über den Datenbus laufenden Daten von 19c nach 17c (Aktiv-Low) geführt werden.

Die Komparatoren 4585 sind dem Baustein 8212 „parallel"-geschaltet. Dies ist auch richtig, da bei direktem Speicherzugriff der 8212 als elektronischer Schalter fungiert, während im Betrieb die Komparatoren den Vergleich zwischen den an den Adreß- und Datenschaltern eingestellten und im Betrieb sich ständig ändernden Adressen und Daten vornehmen. Der Verfasser hat diese Schaltung auf dem Modul für direkten Speicherzugriff mit untergebracht, so daß die Datenschalter mitbenutzt werden.

Soll der Prozessor bei einer bestimmten Adresse oder bei einem bestimmten Dateninhalt angehalten werden, so wird nach Wahl am Schalter S_1, ob Adresse und Daten den Stop auslösen, der Schalter BP geschlossen und das Flipflop durch Drücken der Taste L rückgesetzt. Der Prozessor hält dann bei der gewünschten Adresse oder den gewünschten Daten an. Adresse und Daten werden am Anzeigemodul angezeigt. Da die CMOS-Komparatoren eine Verzögerung bis zu 600 ns pro Baustein haben können, das Ready-Signal aber noch im T_1-Zyklus auf L gehen muß, ist es in jedem Fall erforderlich, mit einem Wait-Zyklus (Verbindung von Anschluß 12a mit 11a über eine Diode) zu arbeiten. Wenn wegen langsamer Speicherbausteine in der Speicherplatine diese Verbindung bereits vorhanden ist, kann sie in Bild 54 entfallen. Es genügt dann ein einpoliger Schalter für BP.

Spannungsüberwachung mit LED-Grenzwertanzeige

Der Mikroprozessor und die anderen Bausteine benötigen konstante Speisespannungen, die nicht mehr als 5 % von den Sollwerten abweichen dürfen. Normalerweise erfüllt das Netzteil diese Anforderungen. Im Störungsfall ist es aber nicht immer leicht mit einem normalen Zeigerinstrument einen Fehler zu erkennen, nämlich z. B. dann, wenn eine Brummspannung überlagert ist. Ferner kommt es nicht selten vor, daß beim Erweitern des Mikrocomputers einmal doch die Stromergiebigkeit der +5 V überschritten werden, wenn man auch noch die externen LED-Anzeigen usw. davon speist. In beiden Fällen schafft man sich einen raschen, zuverlässigen Überblick mit einer LED-Grenzwertanzeige nach *Bild 56*, die auch Wechselstromanteile mit erfaßt. Als Meßglied dient der Fensterdiskriminator TCA 965, der die am Anschluß 8 liegende Spannung mit der durch die Spannungsteiler R1/R2 und R3/R4 heruntergeteilten Referenzspannung (6±0,5 V) vergleicht. Da die Referenzspannung von Exemplar zu Exemplar schwankt, müssen die Teilerwiderstände entweder individuell ausgemessen werden, oder die genaue Anzeige muß mit einem Digitalvoltmeter und Einstellreglern justiert werden. Anschluß 2 zeigt L-Signal unterhalb, Anschluß 13 innerhalb und Anschluß 14 oberhalb des Fensters, welches durch die an Anschluß 7 (untere) und 6 (obere) liegende Spannung gebildet werden. L-Signal läßt die über die Vorwiderstände von 560 Ω angeschlossenen Lumineszenzdioden aufleuchten.

Um die 12 V mit der Referenzspannung besser vergleichen zu können, wird sie mit dem Spannungsteiler 8,2k/2,7k auf 2,97 V heruntergeteilt.

(Fortsetzung folgt)

Bauteileliste für Bild 56

3 Fensterdiskriminatoren TCA 965 (Siemens AG)
6 Lumineszenzdioden LD 30 (Siemens AG)
3 Lumineszenzdioden LD37 (Siemens AG)
3 Widerstände 100 Ω
6 Widerstände 560 Ω
5 Widerstände 2,7 kΩ
2 Widerstände 4,7 kΩ
1 Widerstand 8,2 kΩ
3 Widerstände 10 kΩ
2 Widerstände 12 kΩ
2 Widerstände 22 kΩ
6 Einstellregler 1 kΩ

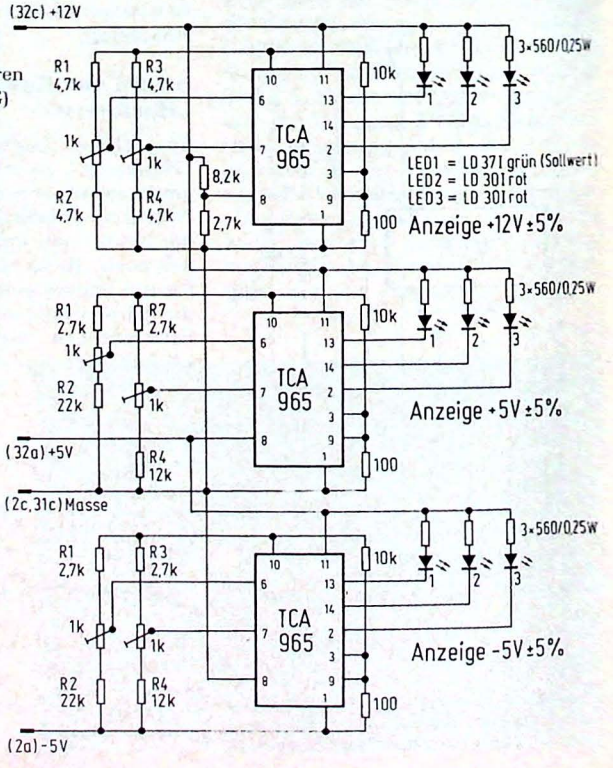


Bild 56. Schaltung zur Spannungsüberwachung der +12 V, +5 V und −5 V-Versorgungsspannungen mit Grenzwertanzeige

Horst Pelka:

Ein elektronisches Würfelspiel läßt sich mit einem Programm von 35 Bytes realisieren. Zur Anzeige der Zahl werden sieben Lumineszenzdioden benötigt.

Praxis mit Mikroprozessoren

19. Teil

Elektronisches Würfelspiel

Sie sind ähnlich wie bei einem Würfel angeordnet (*Bild 57*), nur daß hier die 2 und die 3 nicht diagonal anzeigen, sondern senkrecht. Die mittlere Lumineszenzdiode ist der Eins, die mittleren beiden Rand-Lumineszenzdioden sind der Zwei und die äußeren Lumineszenzdioden der Vier zugeordnet. Entsprechend den Binärzahlen können jetzt aus der 1, 2 und 4 die Zahlen 1 bis 6 zusammengesetzt werden. Die Schaltungen finden Sie in *Bild 58*. Entsprechend der Wertigkeit sind die Lumineszenzdiode für die 1 an die Klemme P_{A0}, die beiden Lumineszenzdioden für die Wertigkeit 2 an die Klemme P_{A1} und die vier Lumineszenzdioden für die Wertigkeit 4 an die Klemme P_{A2} angeschlossen. Da der Ein/Ausgabe-Baustein SAB 8255 nicht vier Lumineszenzdioden gleichzeitig an einer Klemme treiben kann, sind für alle drei Klemmen Emitterfolger vorgesehen, an die dann jeweils ein, zwei oder vier Lumineszenzdioden über Vorwiderstände 150 Ω angeschlossen sind. Ferner ist eine Taste an die Klemme P_{C0} angeschlossen. Bei nicht gedrückter Taste liegt die Klemme über einen Vorwiderstand 4,7 kΩ an H-Pegel, jeweils beim Drücken der Taste liegt dann an der zugehörigen Klemme L-Pegel.

Programm

Wollen wir unsere bisherigen Programme „Seeschlacht" und „Reaktionstest" einschließlich Verzögerungsprogramm beibehalten, so sind die Speicherstellen bis einschließlich Speicherplatz ØØBE besetzt. Das neue Programm könnte also am Platz ØØBF beginnen; diesen lassen wir aber frei und beginnen das Programm erst an der nächsten Stelle mit Platz ØØCØ (*Bild 59*).

Damit der Mikroprozessor an diese Stelle kommt, programmieren wir in den ersten drei Speicherstellen einen Sprungbefehl. Es stehen also auf der Adresse ØØØØ die Daten C3, auf der Adresse ØØØ1 die Daten CØ. Auf der Adresse ØØØ2 kann die ØØ bleiben.

C3 ist der Code für einen unbedingten Sprung, CØ ist das niederwertige Byte der Sprungadresse, die ØØ das höherwertige Byte der Sprungadresse. Bei Verwendung des CMOS-RAMs können wir ganz einfach mit unserem Modul für direkten Speicherzugriff die drei Stellen umprogrammieren. Wollen wir wieder das Spiel „Seeschlacht" und den Reaktionstest benutzen, so sind die ersten beiden Speicherstellen wieder auf ØØØØ zurückzuprogrammieren.

Steht das Programm in einem REPROM, wird das C-MOS-RAM für den Sprung zuhilfe genommen.
Das geschieht folgendermaßen:

- Umschalter (Bild 22) auf C-MOS-RAM als Programmspeicher
- Adresse wie oben einlesen
- Schalter Einzelschritte auf „Single Step" (Bild 3)
- Reset-Taste drücken
- Taste „Step" 3mal drücken, bis Adresse ØØCØ angezeigt wird
- Umschalter (Bild 22) auf REPROM als Programmspeicher
- Schalter Einzelschritte öffnen

Damit befinden wir uns im Programmbereich des Würfelspieles.

In *Bild 60* finden wir das Flußdiagramm. Mit den ersten beiden Operationen wird das Steuerwortregister des Ein/Ausgabe-Bausteines SAB 8255 programmiert. Dann folgt eine Schleife, ob die Taste gedrückt wurde. Wurde die Taste gedrückt, wird in das Register B die Zahl ØØ eingeschrieben. Es folgen dann zwei ineinandergreifende Schleifen. Bei jedem Durchlauf wird das Register B um eins erhöht. In der größeren Schleife wird jedesmal die Taste abgefragt. War diese Taste nicht mehr gedrückt, wird der Akkumulator mit dem Inhalt des Registers B geladen. Dieser Inhalt kann sich zwischen 1 und 6 bewegen. In einer Ausgabeoperation wird der Inhalt des Akkumulators auf dem Kanal A ausgegeben, als nächstes folgt ein Sprung wieder zur Abfrage der Taste.

Eine kleinere Schleife bewirkt, daß das Register nur bis sechs zählen kann. Hat nämlich das Register B den Inhalt sieben bekommen, bewirkt ein Vergleich und anschließender Sprungbefehl einen Sprung zur Speicherstelle ØØCB. Der dort stehende Befehl lädt das Register wieder mit ØØ, da bei dem Würfelspiel nur Ziffern zwischen eins und sechs vorkommen dürfen.

Programmbeschreibung

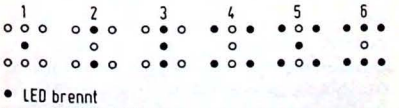
Mit dem ersten Befehl wird die Sedezimalzahl 81 in den Akkumulator geladen. Er steht auf
Adresse ØØCØ mit Daten 3E und
Adresse ØØC1 mit Daten 81.

Als nächstes wird der Inhalt des Akkumulators in das Steuerwortregister des Ein/Ausgabe-Bausteines geschrieben. Die Adresse des Steuerwortregisters ist EF. Dieser Befehl steht auf
Adresse ØØC2 mit Daten DB und
Adresse ØØC3 mit Daten EF.
Die Ausgabe erfolgt auf
Adresse EFEF mit Daten 81.

Das Steuerwort 81 H bedeutet, daß der Ein/Ausgabe-Baustein im Mode Ø arbeitet, wobei die Kanäle A und B als Ausgabe-Kanäle, der Kanal C als Eingabe-Kanal konfiguriert ist. Der Kanal C hat die Adresse EE. Als nächstes wird abgefragt, ob die Taste TA_0 gedrückt ist.

Bauteileliste

3	Transistoren BC 238	(Siemens AG)
7	Lumineszenzdioden LD 52	(Siemens AG)
7	Widerstände 150 Ω	
1	Widerstand 4,7 kΩ	
1	einpolige Taste	



● LED brennt
○ LED brennt nicht

Bild 57. Zusammensetzung der Würfelzahlen 1 bis 6 aus den sieben Leuchtpunkten

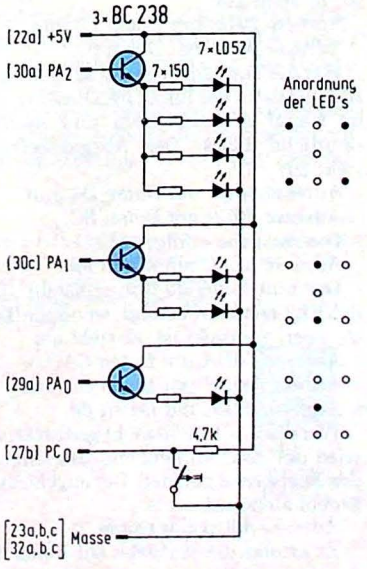


Bild 58. ▶
Anschaltung der Lumineszenzdioden und Tasten an den Ein-Ausgabebaustein SAB 8255. Rechts: Zuordnung der LED zu den Leuchtpunkten

Dies erfolgt mit einer Eingabe, der zugehörige Befehl steht auf
Adresse ØØC4 mit Daten DB und
Adresse ØØC5 mit Daten EE.
Es erfolgt nun die eigentliche Eingabe auf
Adresse EEEE mit Daten Ø1,
wenn die Taste nicht gedrückt ist.

Dies kommt daher zustande, weil normalerweise die Klemme P_{C0} auf H-Pegel liegt, also die äquivalente Binärzahl Ø1 angezeigt wird. Das Herausfinden, ob die Taste gedrückt ist, erfolgt mit dem Maskierungsbefehl ANI Ø1 H. Er steht auf
Adresse ØØC6 mit Daten E6 und
Adresse ØØC7 mit Daten Ø1.

Ist die Taste TA_0 nicht gedrückt, bleibt im Akkumulator an der niederwertigsten Stelle ein H stehen. Der Inhalt des Akkumulators ist also nicht Ø, der nachfolgende Sprungbefehl „Springe, wenn der Inhalt des Akkumulators nicht Null ist" („JNZ Start") führt das Programm dann wieder auf die Speicherstelle ØØC4 zurück. Dieser Sprungbefehl steht auf.
Adresse ØØC8 mit Daten C2,
Adresse ØØC9 mit Daten C4 und
Adresse ØØCA mit Daten ØØ.

Bei dieser Gelegenheit soll daran erinnert werden, daß Adressen beim System 8080 immer zuerst mit dem niederwertigen Byte und dann mit dem höherwertigen Byte geschrieben werden. War die Taste gedrückt, wird der letzte Sprungbefehl nicht ausgeführt; das Programm geht also weiter zu
Adresse ØØCB mit Daten Ø6 und
Adresse ØØCC mit Daten Ø1.

Dieser Befehl bewirkt, daß in das Register B die Zahl ØØ eingeschrieben wird. Der nächste Befehl erhöht die im Register B stehende Zahl jeweils um eins. Er steht auf
Adresse ØØCD mit Daten Ø4.

Nunmehr muß sichergestellt werden, daß die Erhöhung nur bis Ø6 erfolgt, es wird also das Register B mit der im Akkumulator stehenden Zahl Ø7 verglichen. Ergibt dieser Vergleich eine Übereinstimmung, erfolgt ein Rücksprung zur Adresse ØØCB, auf der der Befehl zum Einschreiben der ØØ in Register B steht. Dieser Vergleich umfaßt zwei Befehle, nämlich einmal das Einschreiben in den Akkumulator und das zweite Mal den eigentlichen Vergleichsbefehl „CMP B". Diese Befehle stehen auf
Adresse ØØCE mit Daten 3E,
Adresse ØØCF mit Daten Ø7,
Adresse ØØDØ mit Daten B8.

Der nun folgende bedingte Sprungbefehl „JZ EINS" steht auf
Adresse ØØD1 mit Daten CA,
Adresse ØØD2 mit Daten CB und
Adresse ØØD3 mit Daten ØØ.

```
ISIS  8080   MACRO ASSEMBLER,  V1.1 PAGE 1


ØØCØ                          ORG    ØØCØ
ØØCØ  3E81          WUERF:    MVI    A,81 H
ØØC2  D3EF                    OUT    EF
ØØC4  DBEE          START:    IN     EE
ØØC6  E6Ø1                    ANI    Ø1H
ØØC8  C2 C4ØØ                 JNZ    START
ØØCB  06ØØ          EINS:     MVI    B,ØØ
ØØCD  Ø4            INKRE:    INR    B
ØØCE  3EØ7                    MVI    A,Ø7H
ØØDØ  B8                      CMP    B
ØØD1  CACBØØ                  JZ     EINS
ØØD4  DB EE         STOP:     IN     EE
ØØD6  E6Ø1                    ANI    Ø1H
ØØD8  D3EC                    OUT    EC
ØØDA  CACDØØ                  JZ     INKRE
ØØDD  78                      MOV    A,B
ØØDE  D3EC                    OUT    EC
ØØEØ  C3C4ØØ                 JMP    START
ØØCØ                          END
```

▲ Bild 59. Programm für das Würfelspiel. Links steht der Maschinencode, rechts der mnemonische Assemblercode

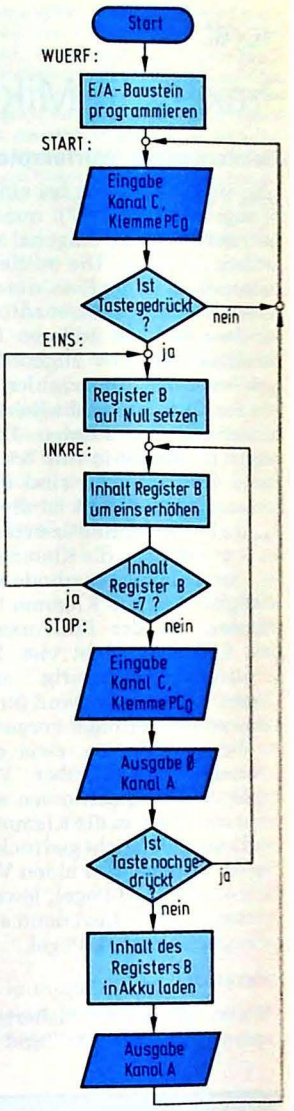


Bild 60. ▶
Flußdiagramm für das Würfelspiel

Es folgt nunmehr eine Eingabe, bei der abgefragt werden soll, ob die Taste noch geschlossen ist. Die Eingabe steht auf
Adresse ØØD4 mit Daten DB und
Adresse ØØD5 mit Daten EE.
Es folgt der Eingabevorgang auf
Adresse EEEE mit Daten Ø1.
wenn die Taste losgelassen ist, bzw. ØØ, wenn die Taste noch gedrückt ist. Der Maskierungsbefehl „ANI Ø1 H" stellt fest, ob die Taste noch gedrückt ist. Er steht auf
Adresse ØØD6 mit Daten E6 und
Adresse ØØD7 mit Daten Ø1.

War die Taste noch gedrückt, steht im Akkumulator der Inhalt ØØ. Dieser wird im Kanal A ausgegeben und löscht sämtliche LEDs. Der Ausgabebefehl steht auf
Adresse ØØD8 mit Daten D3 und
Adresse ØØD9 mit Daten EC.
Die Ausgabe erfolgt auf
Adresse ECEC mit Daten ØØ.

Der nun folgende Sprungbefehl „JZ INKRE" wird nur befolgt, wenn die Taste noch gedrückt ist. Er steht auf
Adresse ØØDA mit Daten CA,
Adresse ØØDB mit Daten CD und
Adresse ØØDC mit Daten ØØ.

War die Taste nicht mehr gedrückt, so wird der Akkumulator mit dem Inhalt des Registers B geladen. Der zugehörige Befehl steht auf
Adresse ØØDD mit Daten 78.

Es erfolgt die Ausgabe auf Kanal A, der die Adresse EC hat. Der Befehl steht auf
Adresse ØØDE mit Daten D3 und
Adresse ØØDF mit Daten EC.

Die Ausgabe auf Adresse ECEC kann jetzt Daten zwischen Ø1 und Ø6 haben. Dies hängt vom Zufall ab, zu welchem Zeitpunkt die Taste losgelasen wurde, also welchen Inhalt jetzt unser Würfel ausgibt. Um wieder neu würfeln zu können, erfolgt ein unbedingter Sprung zur Startadresse ØØC4. Dieser Befehl steht auf
Adresse ØØEØ mit Daten C3,
Adresse ØØE1 mit Daten C4 und
Adresse ØØE2 mit Daten ØØ.

Vergleichsweise finden Sie in [8] einen elektronischen Würfel, der mit festverdrahteter Logik realisiert ist.

(Fortsetzung folgt)

Literatur

[8] *Ruff, A.*: Elektronische Unterhaltungsspiele. Franzis-Verlag

Horst Pelka:

In Teil 6 (Heft 20/1977, Seite 946 dieser Reihe) war angekündigt worden, daß auch ein anschlußkompatibler Modul mit dem Mikroprozessor 8085 in dieser Serie beschrieben wird. Mit dem 8085 vereinfacht sich auch die Schaltung.

Praxis mit Mikroprozessoren

20. Teil

Der neue Mikroprozessor 8085

Der 8085 ersetzt drei Bausteine des 8080-Systems: den Mikroprozessor 8080, den Taktgeber und Treiber 8224 und den System-Steuer- und Bus-Treiber-Baustein 8228 (*Bild 61*). Er ist weitgehend „Software"-kompatibel zum 8080. Während beim 8080 ein getrennter Adreß- und Datenbus vorhanden ist, besitzt der 8085 ein für die niederwertigen 8 Bit gemultiplexten Adreß- und Datenbus. Für die höherwertigen 8 Bit ist der Adreßbus nach wie vor getrennt (*Bild 62*).

Minimalsysteme werden daher mit speziellen Ein/Ausgabe- und Speicherbausteinen aufgebaut. *Bild 63* zeigt die Prinzipschaltung eines solchen Minimalsystems. Zusätzlich zum Mikroprozessor 8085 wird ein Baustein 8355/8755 und ggf. ein Baustein 8155 benötigt. Der 8355/8755 enthält 2 k ROM/PROM und zwei Ein/Ausgabe-Kanäle mit einer Breite von 8 Bit. Innerhalb des Bausteines werden die Adressen und Daten demultiplext; zur Synchronisation erhält der Baustein ein Signal ALE über den Steuerbus. Dieses Signal steht während des T_1-Zyklus an (*Bild 64*). Die zu diesem Zeitpunkt anliegende Adresse wird zwischengespeichert und steht bei den nachfolgenden T-Zyklen zur Verfügung. Der Baustein besitzt auch eine eingebaute WAIT-Schaltung.

Werden in diesem Minimalsystem auch noch Schreib/Lese-Speicher benötigt, wird der Baustein 8155 eingesetzt. Er enthält zwei weitere Kanäle zu 8 Bit und einen Kanal zu 6 Bit, ferner einen programmierbaren Zähler. Sind für die Aufgabe jedoch die eingebauten Register ausreichend, so ist das Minimalsystem bereits mit dem 8085 und dem 8355/8755 funktionsfähig.

Neu gegenüber dem 8080 sind vier Eingänge für Unterbrechungen mit unterschiedlichen Prioritätsebenen. Darauf werden wir aber später noch zurückkommen. Während beim 8080-System die Zustandsinformation zu Beginn des T_2-Zyklus über den Datenbus gemultiplext wurde (siehe Bild 31 in Teil 7), erfolgt die Zustandsinformation beim 8085 über die Statusausgänge S_0 und S_1. *Bild 65* zeigt die Übersicht der Zustandsinformationen bei verschiedenen Pegeln an diesen Ausgängen im Zusammenhang mit den Pegeln am Steuerbus. *Bild 66* zeigt die Zustände an den Klemmen S_0 und S_1, am gemultiplexten Adreß-Datenbus und an drei Kontrollbusleitungen bei den verschiedenen T-Zyklen.

Für den Aufbau erweiterungsfähiger Mikroprozessorsysteme, zu denen auch das hier beschriebene System gehört, ist es wirtschaftlicher, Standardspeicher und Standard-Ein/Ausgabe-Bausteine einzusetzen. Der nachfolgend beschriebene Modul ist für solche Standard-Bausteine ausgelegt und läßt sich direkt mit dem in Teil 6 und Teil 7 besprochenen Mikroprozessor-Modul austauschen.

Mikroprozessor-Modul mit 8085

Kern des Mikroprozessor-Moduls in *Bild 66* ist der Mikroprozessor 8085. Da der Taktgeber im Baustein integriert ist, kann der Quarz direkt angeschlossen werden. Zum einwandfreien Schwingen muß noch von jedem Anschluß des Quarzes je ein Kondensator von 20 pF gegen Masse geschaltet werden. Zum Demultiplexen der niederwertigen 8 Bit des Adreßbusses wird der 8-Bit-Eingabe/Ausgabe-Baustein 8212 verwendet. Dieser zusätzliche Baustein stellt insbesondere bei später erweitertem System keinen zusätzlichen Aufwand dar, wenn der Fan-out des Adreßbusses beim Anschluß vieler Speicher sowieso erweitert werden muß. In diesem Falle, wird der gleiche Baustein auch noch einmal als Puffer für die höherwertigen 8 Bit des Adreßbusses verwendet.

Vom Mikroprozessor 8085 führt die Steuerleitung ALE auf den STROBE-Übernahmeeingang STB des Bausteines 8212. Bei der negativen Flanke des ALE-Impulses wird die am kombinierten Adreß-Datenbus stehende Adresse in den aus D-Flipflops bestehenden Zwischenspeichern des 8212 übernommen. Die Adresse steht also bis zur nächsten negativen Flanke des ALE-Impulses an den Adreßleitungen A_0 bis A_7 zur Verfügung.

Weiterhin enthält die Mikroprozessorplatine einen Baustein FLH 211 (7404, sechsfacher Inverter) und den Baustein FLH 201 (7401, vier NAND-Glieder mit je zwei Eingängen und offenem Kollektor). Zwei Inverter stellen die richtige Polarität des $\overline{\text{HOLD}}$- und des $\overline{\text{INT}}$-Signales am Mikroprozessor her, da dieser H-aktive Signale braucht. Drei Inverter und vier NAND-Glieder werden zur Decodierung der auf unserer Rückwandverdrahtung genormten Signale $\overline{\text{MEMR}}$, $\overline{\text{MEMW}}$, $\overline{\text{I/OR}}$ und

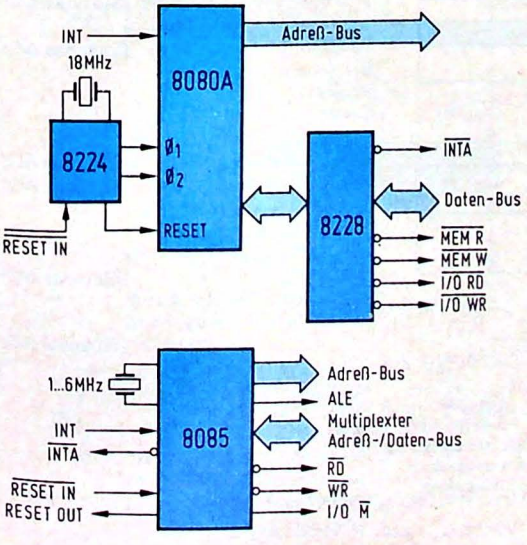


Bild 61. Unterschied zwischen der CPU mit 8080 und 8085

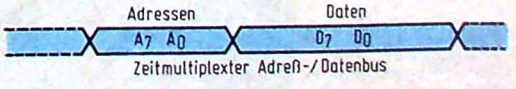


Bild 62. Zeitdiagramm des kombinierten Adreß-/Datenbusses beim Mikroprozessor 8085

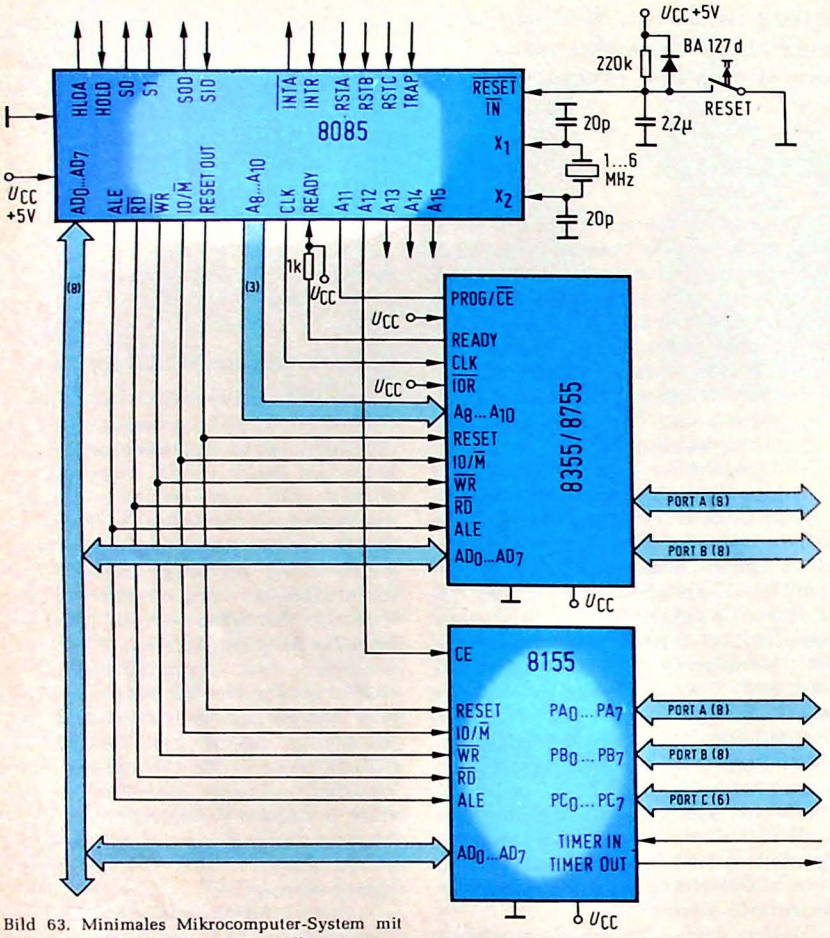


Bild 63. Minimales Mikrocomputer-System mit 8085 unter Verwendung von Spezialbausteinen

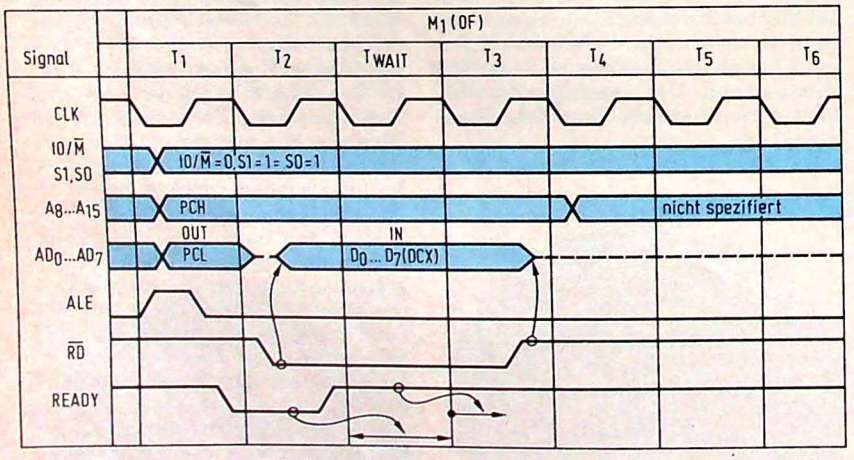


Bild 64. Zeitdiagramm eines „Befehlshole-Maschinen-Zyklus" beim 8085

Maschinen-Zyklus		Status IO/M	Status S1	Status SO	Steuersignale RD	Steuersignale WR	Steuersignale INTA
Befehlsholezyklus	(OPCODE FETCH) (OF)	L	H	H	L	H	H
Speicher lesen	(MEMORY READ) (MR)	L	H	L	L	H	H
Speicher einschreiben	(MEMORY WRITE) (MW)	L	L	H	H	L	H
Eingang lesen	(I/O READ) (IOR)	H	H	L	L	H	H
Ausgang schreiben	(I/O WRITE) (IOW)	H	L	H	H	L	H
Unterbrechungsanfragequittung	(INTR ACKNOWLEDGE) (INA)	L	H	H	H	H	L
Bus nicht benutzt	(BUS IDLE) (BI)	X	X	X	H	H	H

X = unbestimmt

Bild 65. Status-Information und Steuerbuspegel bei verschiedenen Maschinenzyklen

I/OW gebraucht, da der Mikroprozessor an seinen Anschlüssen 31, 32, 34 die Signale WR, RD und IO/M führt.

Im Gegensatz zum 8080-System braucht der Mikroprozessor 8085 nur eine Betriebsspannung von +5 V.

Der 8085 kann T-Zyklen von 320...2000 ns verarbeiten. Die vom Taktgenerator erzeugte Frequenz wird intern noch einmal um den Faktor zwei geteilt. So entsprechen diese Zeiten Quarzfrequenzen von 1...6 MHz. Um zeitkompatibel mit dem in Teil 6 und Teil 7 beschriebenen Mikroprozessor-Modul des 8080 zu sein, muß ein Quarz von 4,096 MHz verwendet werden. Besteht der Wunsch für eine schnellere Verarbeitung, kann an dieser Stelle ein 6-MHz-Quarz eingesetzt werden, nur müssen die Programme, die Zeitfunktionen enthalten, dann entsprechend umgeschrieben werden.

(Schluß folgt)

T-Zyklus	Status & Busse S1,SO	IO/M A8...A15	AD0...AD7	Steuersignale RD,WR	INTA	ALE
T1	X	X	X	H	H	H
T2	X	X	X	X	X	L
TWAIT	X	X	X	X	X	L
T3	X	X	X	X	X	L
T4	H	X	TS	H	H	L
T5	H	X	TS	H	H	L
T6	H	X	TS	TS	H	TS
TRESET	H	TS	TS	TS	H	TS
THALT	L	TS	TS	TS	H	TS
THOLD	X	TS	TS	TS	H	TS

TS = Three-State im hochohmigen Zustand
X = unbestimmt

Bild 66. Zustände an den Klemmen S_0 und S_1 am kombinierten Adreß-Datenbus und am Steuerbus während der einzelnen T-Zyklen

Bauteileliste

Zu Bild 63

1 Mikroprozessor SAB 8085 (Siemens AG)
1 8-Bit-Eingabe/Ausgabe-Baustein SAB 8212 (Siemens AG)
Bei Bedarf ein weiterer Baustein SAB 8212
1 Baustein 4 NAND-Glieder mit je zwei Eingängen und offenem Kollektor FLH 201 (Siemens AG)
1 6facher Inverter FLH 211 (Siemens AG)
1 Schwingquarz 4,096 MHz
2 Styroflex-Kondensatoren 22 pF, 63 V Gleichspannung B 31310-A 5220 – F (Siemens AG)
1 Keramikkondensator 0,1 µF, 63 VB37449 – A6104 – S 02 (Siemens AG)
1 Tantal Perlkondensator 2,2 µF, 30 V
1 Tantal Perlkondensator 10 µF, 15 V
2 Widerstände 1 kΩ, 0,25 W
3 Widerstände 4,7 Ω, 0.25 W
1 Widerstand 220 kΩ, 0,25 W
1 einpolige Taste

Horst Pelka

Mit dem nun folgenden Teil ist unsere Reihe „Praxis mit Mikroprozessoren" beendet. Das soll nun nicht heißen, daß Sie in der FUNKSCHAU ab sofort nichts mehr über dieses Thema lesen – warten Sie ab...

Praxis mit Mikroprozessoren

21. Teil und Schluß

Soll das in Folge 11 beschriebene Unterbrechungsprogramm vollkompatibel laufen, muß die oben in Bild 66 in Strichen gezeichnete Zusatzschaltung mit einem Inverter und 8 Dioden BA 127 D eingebaut werden, um bei einer Unterbrechungsanforderung auf die RST-7--Adresse zu kommen. Diese Adresse ist im Hexcode 38. Nach einer Unterbrechungsanforderung an Anschluß 10 sendet zu gegebener Zeit Anschluß 11 einen INTA-Impuls aus. Dieser Impuls wird durch den Inverterter in einen positiven Impuls umgewandelt, zum Zeitpunkt ist der Datenbus auf Eingabe geschaltet, die 8 Dioden BA 127 legen an jede Datenbusleitung H-Pegel an, was dem hexadezimalen Maschinencode FF entspricht. Dieser Code führt einen Sprung zur Adresse 38 herbei (siehe Teil 10, Tabelle 1 und 2).

Eine andere Möglichkeit ist, die Interruptanforderungen an einen von den Anschlüssen 7 bis 9 heranzuführen und die RST-Adressen 5,5, 6,5 bzw. 7,5 zu wählen. In diesem Falle sollte zwischen dem Ausgang des Inverters, der an Anschluß 19 a angeschlossen ist, ein Wahlschalter sein, der entweder zu Anschluß 10 oder 7 bis 9 führt. An ihm kann dann eingestellt werden, welche RST-Adresse man wünscht (*Bild 67*). Gleichzeitig muß das Programm entsprechend umgeschrieben werden.

H-Pegel an RST 5,5 löst einen Sprung zur Adresse 2 C.

H-Pegel an RST 6,5 löst einen Sprung zur Adresse 34 und

H-Pegel an RST 7,5 löst einen Sprung auf Adresse 3 C aus

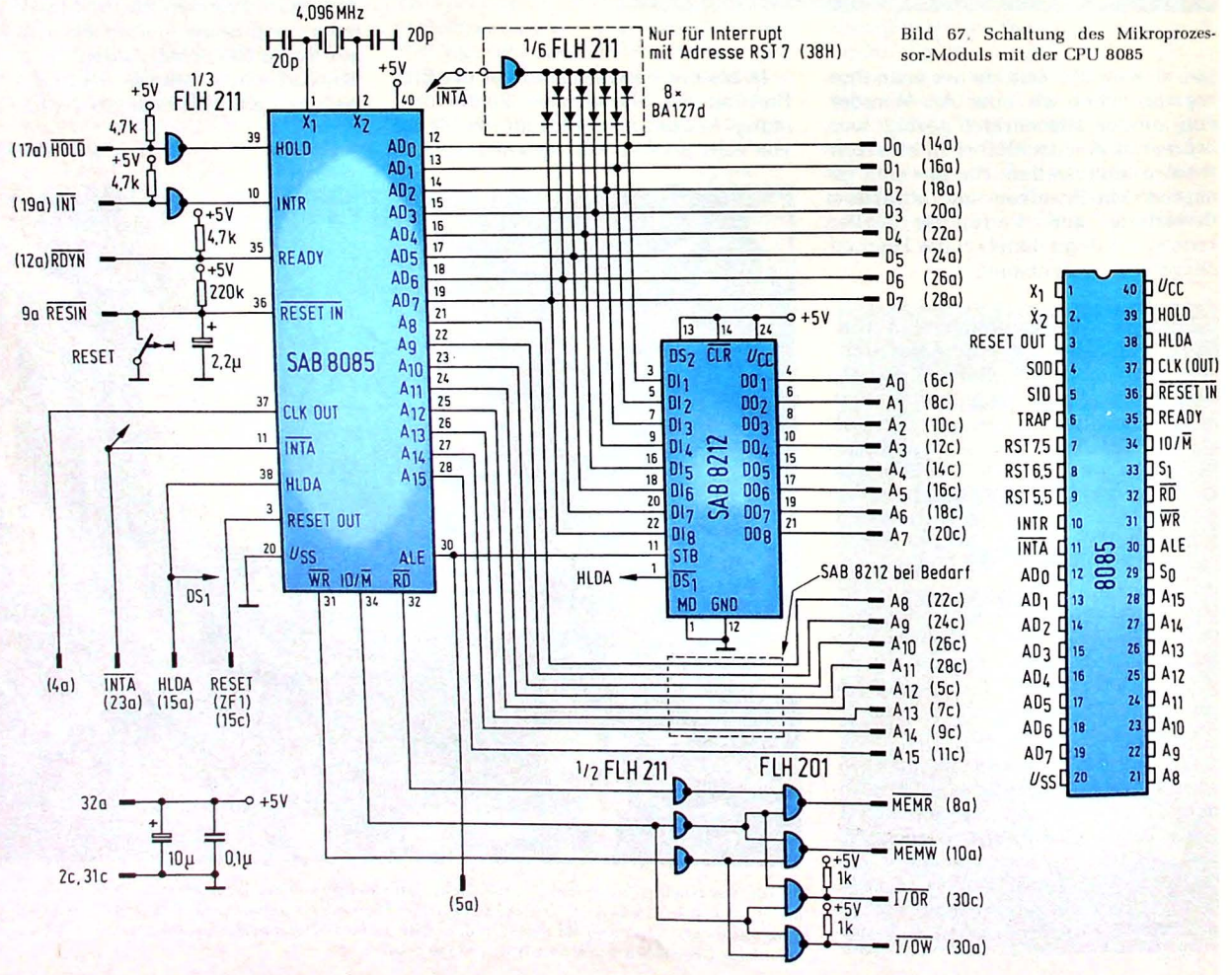


Bild 67. Schaltung des Mikroprozessor-Moduls mit der CPU 8085

RST 5,5 und RST 6,5 sprechen auf H-Pegel ähnlich wie der INTR-Eingang an, während RST 7,5 auf die ansteigende Flanke triggert. Daher ist für RST 7,5 nur ein schmaler Impuls zur Unterbrechungsanfrage notwendig. Jeder der drei RST-Anschlüsse kann individuell durch den SIM-Befehl maskiert werden. Die entsprechende Wahrheitstabelle finden Sie in *Bild 69*. Das

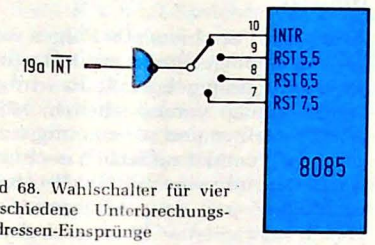


Bild 68. Wahlschalter für vier verschiedene Unterbrechungs-Adressen-Einsprünge

entsprechende Muster muß vorher in den Akkumulator geladen werden. Je nach dem, in welche RST-Adresse man nun die Unterbrechungsanforderung eingibt, muß das Programm entsprechend geändert werden. Dazu muß der Sprungbefehl, der die Ansprungadresse des Unterprogramms überbrückt und 3 Bytes lang ist, drei Adressen vorher stehen. Auf der RST-Adresse steht dann der Sprungbefehl zu dem Unterbrechungs-Unterprogramm. Die ausgelassenen Befehle (beim RST 5,5 und RST 6,5) müssen dann hinterher untergebracht werden, bzw. bei RST 7,5 vorher.

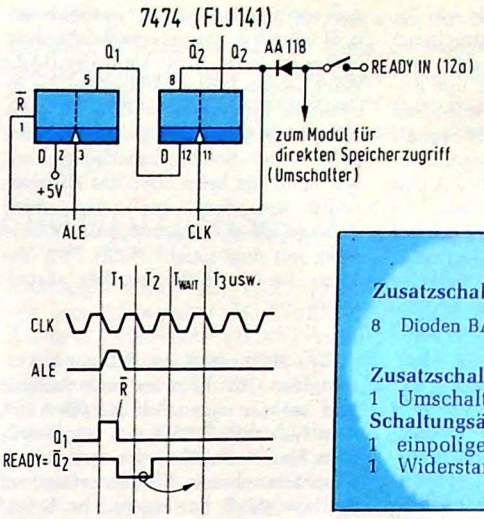


Bild 70. Schaltung zur Erzeugung eines Warte-(WAIT-)Zyklus beim 8085

Zusatzschaltung für RST-7-Adresse

8 Dioden BA 127 D (Siemens AG)

Zusatzschaltung nach Bild 67
1 Umschalter mit vier Stellungen
Schaltungsänderung nach Bild 61
1 einpoliger Umschalter
1 Widerstand 4,7 kΩ

Der entsprechende Reset-Befehl für die Maske kann auf den Adressen 0000 bis 0002 untergebracht werden. Wichtig ist, daß dieser vor dem „EI"-Befehl steht. Für das Rücksetzen der Masken für alle drei Unterbrechungsanfrageeingänge lautet er:

```
0000 3E08   MVI A,08
0002 30     SIM
```

Schaltung für WAIT-Zyklus

Zum Anschluß von langsamen Speichern konnte beim 8080-System durch Verbinden der Klemmen READY und WAIT ein WAIT-Zyklus eingeführt werden. Das Programm für den Reaktionstest war so geschrieben worden, daß es für einen WAIT-Zyklus die richtigen Zeiten programmierte, damit man billige CMOS-Speicher einsetzen konnte. Der 8085 hat keinen getrennten WAIT-Ausgang, so daß man beim Anschluß von Standard-Speichern, die keinen READY-Ausgang besitzen, eine Zusatzschaltung mit zwei D-Flipflops nach *Bild 70* braucht. Diese Zusatzschaltung funktioniert folgendermaßen.

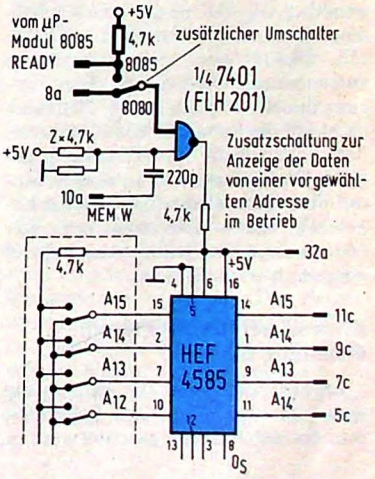
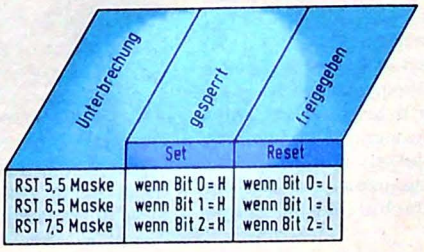


Bild 71. Anbringung eines Umschalters zur Zeitanpassung des Komparators im Modul für direktem Speicherzugriff für 8080- und 8085-Systeme



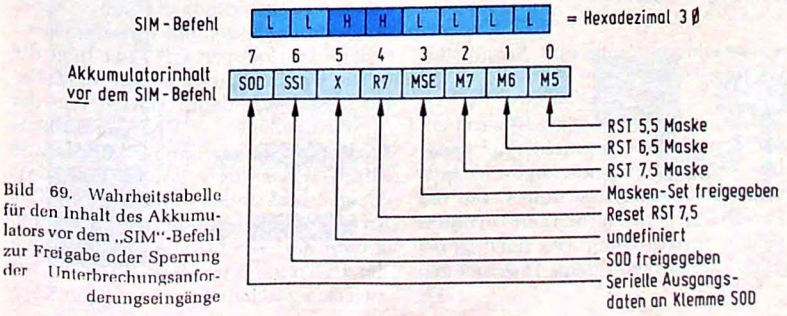


Bild 69. Wahrheitstabelle für den Inhalt des Akkumulators vor dem „SIM"-Befehl zur Freigabe oder Sperrung der Unterbrechungsanforderungseingänge

Der ALE-Impuls schiebt die am D-Eingang liegenden 5 V (entsprechend H-Pegel) an den Ausgang Q_1 durch. Das zweite D-Flipflop übernimmt mit ansteigender Flanke des Clock-Impulses an seinem D-Eingang dieses H-Signal, sein Ausgang Q_2 geht auf L-Pegel. Dieser L-Pegel wird über die Diode AA 118 an die RDYN-Leitung (Anschluß 12 a) gegeben. Gleichzeitig führt Q_2 auf R des ersten Flipflops und setzt diesen rück. Während des T_2-Zyklus akzeptiert der Mikroprozessor den L-Pegel an seinem READY-Eingang und führt einen T_{WAIT}-Zyklus herbei.

Wie wir später noch sehen werden, wird dieser negative READY-Impuls beim 8085 auch noch zur Anzeige von Daten und Adressen während des Betriebes und die Warteschleife für eine PROM-Programmier-Platine benötigt. Es ist also empfehlenswert, diese Schaltung auch dann vorzusehen, wenn keine langsamen Speicher angeschlossen sind.

Anzeige der Daten einer vorgewählten Adresse im Betrieb

Gegenüber dem 8080 sind die Zeitabläufe im 8085 teilweise verschieden, z. B. werden die niederwertigen Bits der Adresse und die Daten auf dem Datenbus multiplext. Daher muß bei Verwendung des 8085 in der Zusatzschaltung (siehe dritter Teil, Heft 17, Seite 115, Bild 17) eine kleine Änderung vorgenommen werden. Der obere Eingang des NAND-Gatters FLH 201 führt nicht auf die Leitung MEM R (8a) sondern wird auf die READY-Leitung geführt. Der Verfasser hat an seinem Modul dort einen kleinen Umschalter angebracht, der je nach Einsetzen einer 8080- oder 8085-Platine entsprechend eingestellt wird (Bild 61).

Hinweis bei der Programmerstellung für 8085

Grundsätzlich sind die Befehle des 8080 und 8085 kompatibel. Beim 8085 muß besonders darauf geachtet werden, daß vor dem DAA-Befehl das Hilfscarry-Bit durch einen anderen Befehl nicht verändert wird, sonst kann der DAA-Befehl einen Fehler hervorrufen. Da in unserem Programm in Bild 46 nach Adresse ØØ9Ø bis zum DAA auf Adresse ØØ99 einige Befehle zwischengeschoben sind, die beim 8085 das Hilfscarry-Bit verändern, muß man nach Adresse ØØ9Ø zunächst auf Adresse ØØ91 mit dem Befehl PUSH PSW die Flags in den Kellerspeicher abspeichern.

Die nachfolgenden Adressen verschieben sich dann um eine Adresse und auf der neuen Adresse ØØ9A mit dem Befehl POP PSW den ursprünglichen Flag-Zustand wieder aus dem Kellerspeicher holen. Sodann erfolgt auf Adresse ØØ9B der eigentliche Befehl DAA, alle weiteren Adressen verschieben sich um zwei Stellen, so auch die Anzeigestelle ØØAF auf ØØBl. Dies ist auf den Adressen ØØ75 und ØØ9D (neu). Dies ist auch beim CALL-Befehl auf Adresse ØØ78 zu berücksichtigen, wo an Stelle der BØ jetzt eine B2 stehen muß.

Literatur

Siemens-System SAB 8085, Bestell-Nr. B 1787

Hinweis

Die bis heute erschienenen Folgen werden in dem Buch „Praxis mit Mikroprozessoren" zusammengefaßt. Es wird im Franzis-Verlag voraussichtlich Mitte 1978 erscheinen und neben einigen Ergänzungen enthält zusätzlich noch eine Schaltung für eine PROM-Programmierplatine mit zugehörigem Programm sowie eine Zusatzschaltung zum Überspielen von Programmen auf Kassettenrecorder mit zugehörigem Programm. ■

Hf-Entstörmittel in neuer Ausführung

In neuer, völlig Hf-dichter Metallgehäuseausführung und dadurch auch mit vergrößertem Schirmmaß entsprechend den „Technischen Vorschriften für Rundfunk-Empfangs-Antennenanlagen" der Bundespost liefert Philips den Hf-Trenntransformator Hf/Tr 7104 und den Hf-Hochpaß HP 7104. Beide haben das gleiche Gehäuse und Koaxialsteckverbinder (*Bild*), so daß sie zwischen den Antennenanschluß nach DIN 45 325 des Fernsehempfängers und das Antennenanschlußkabel gesteckt werden können.

Der Hf-Trenntransformator Hf/Tr 7104 verhindert Fernsehempfangsstörungen im Frequenzbereich bis 800 MHz durch örtliche Rundfunksender, Kurzwellenstationen und KW-Amateurfunkstellen, die auf Mantelströme der Antennenleitung (Einstrahlung auf die Abschirmung des Koaxialkabels) zurückzuführen sind. Aber auch die Einstrahlung von Oberwellen des Zeilenoszillators und des Switchmode-Netzteils der Fernsehempfänger in das Kabelnetz von Gemeinschaftsantennen-Anlagen, die Anlaß zu Störungen des Empfangs angeschlossener Tonrundfunkempfänger sein kann, wird damit weitgehend unterbunden.

Der Hf-Trenntransformator Hf-Tr 7104 und der Hf-Hochpaß HP 71094 haben das gleiche, völlig Hf-dichte Metallgehäuse

Beim Hf-Hochpaß HP 7104 liegt die untere Grenzfrequenz bei 40 MHz. Daher lassen sich damit starke, von der Antennenanlage aufgenommene Fremdsignale von Lang-, Mittel- und Kurzwellensendern sowie von KW-Amateurfunkstellen unterdrücken. Treten außerdem Störungen durch Mantelströme auf, so wird zusätzlich noch der Hf-Trenntransformator Hf/Tr 7104 zwischengeschaltet. *Egon Koch*

TM 990-Handbuch jetzt in Deutsch

Für das Mikrocomputer-Lernsystem, bestehend aus dem TM 990/100 M-Single-Board-Computer und dem taschenrechnerähnlichen Mikroterminal TM 990/301, ist nun auch ein deutschsprachiges Handbuch lieferbar, das auch einzeln erhältlich ist. Es behandelt die Themen TIBUG-Befehle und Hilfsprogramme, Befehlssatz des Mikroprozessors TMS 9900, Programmiertechniken, Aufbau der Computerkarte, Speichererweiterung und Schnittstellen-Realisierung.

Damit ist es auch dem „Newcomer" auf dem Mikrocomputer-Sektor möglich, sich ohne perfekte Englischkenntnisse (die sonst meist zum Lesen der Handbücher erforderlich sind) in dieses Gebiet einzuarbeiten. Das Buch ist bei allen Texas-Instruments-Distributoren erhältlich. Fe.